Faculty of Computer Science and Information Technology

University of Malaya

Virtual Auctions

Prepared by : **Louis Tan Han Chooi**
WEK 990052
Supervisor : **Dr. Lee Sai Peck**
Moderator : **Mr. Chiew Thiam Kian**

ABSTRACT

Virtual Auctions is about developing an e-auction system for online auctioning off general merchandise to bidders in specialized auctions such as travel auction, computer auction, car auction, etc. The system will allow bidders to do online registration and bidding, which could be for auctioning off many interesting, bizarre and exciting items such as watches, antiques, etc.

In the development of this system, there are several stages involved, which we will see in the following chapters of this report. Firstly there will be an Introduction chapter where the objectives and scope of this project is determined. Also there is a part where the system development approach will be explained together with the methodology to be used in the development of the Virtual Auctions system. Then a Literature Review is done in the second chapter. The purpose of literature review is to gather as much relevant information as possible about the Virtual Auctions project. It is very important as it recognizes other projects that might have similar characteristics to this project. Therefore, a literature review will be able to help the developer know the features offered by existing systems.

After gathering relevant information about the project in the Literature Review, we move on to the Analysis chapter where the systems analysis is done. The systems analysis stage is where all the requirements of the Virtual Auctions system are analyzed and a requirements model is produced as a result. Then the next chapter will be the Design chapter where the proposed system will be designed according to the requirements specifications done in the analysis phase.

It is hoped that the development of this system will give much invaluable experience and knowledge in the systems development field. It is also a vision for this system to achieve its objectives and to provide a better solution to the local web-based auction community.

ACKNOWLEDGEMENTS

In this section, I would like to take the opportunity to acknowledge the effort of some people whose names may not appear anywhere in this report except on this page. However, their hard work, cooperation, friendship and understanding were crucial to the development of the Virtual Auctions system.

First of all, I would like to thank my project supervisor, Assoc. Prof. Dr. Lee Sai Peck, for giving me so much invaluable advice in all the development stages of this project. Without her experience and understanding, I would certainly be grasping for help. Next, I would like to extend my gratitude to Mr. Chiew Thiam Kian who plays the role as my project moderator. I would like to thank him for his patience and understanding especially during the viva phase.

I would also like to thank my fellow Software Engineering course mates for their help and fellowship throughout the duration of this project. Watching them work so hard for the final year project gives me extra motivation even to strive harder in my project.

Last but not least, I would like to thank all those not mentioned here, who in one way or another contributed to the development of the Virtual Auctions system.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1 : INTRODUCTION

1.1 Introduction to Virtual Auctions

The Internet, and especially the service known as the World Wide Web (WWW) has become the communication interface to an opportunity-growing network of computers. The Internet is an immense utility structure for information, communication and media services. Thus it offers an easily accessible interface for electronic commerce (e-commerce).

Using the Internet as a commercial platform is relatively a new development. One of the many diversified activities of e-commerce is online auction. Though the concept of auction has been around for centuries, it has recently become an enormously successful method of e-commerce. Today there are auction sites such as e-Bay (www.ebay.com) and Yahoo! Auctions (auctions.yahoo.com), where all types of commodity items are placed.

In this project, we will go through the phases of developing a similar e-commerce solution for online auction, which will be called Virtual Auctions. The phases involved in developing the Virtual Auctions system will cover System Analysis and Design, System Implementation and Testing.

Throughout the duration of this project, we will work towards the development of several general components in the Virtual Auctions system. These components will include:

- Registration of auctioneers and bidders using secure transaction
- Search facilities for auctioned items on the site
- Bid placement and processing
- Winner declaration and deal processing.

1.2 Objectives

In this section, we will establish the objectives of this project. Objectives are measures of success. It is something that we expect to achieve, if given the sufficient resources. These are the objectives of the Virtual Auctions project:

- To develop an online auction system for general merchandise using relevant development tools.
- To investigate and study the various techniques and skills to develop an auction system in a web based environment.
- To create an object-oriented system to manage all resources and processes involved.
- To implement a simple and user-friendly system that will draw all levels of users in the Internet.
- To guarantee high ability, reliability and confidentiality of the system and to achieve fast response time.
- To create a paperless administration system with easy access to the system's database and also report generation.
- To develop a system that supports future upgrading and reengineering.

1.3 Project Scope

In this project, we will focus on the development of an online based system which will also include administration facilities. The system will also be accompanied by a relevant database system to manage the various data involved. The administration facility should allow entire access to the database system.

Basically, the auction system will have a secured login system for users (auctioneers and bidders) and also administrators. Users can browse through, sell or bid for an item in the auction system.

1.4 Aims, Relevance and Significance

1.4.1 Aims

The aim of this project is to develop a fully functional online auction system. The final product shall be able to meet all the objectives of this project. In general, it shall also fulfil the fundamental needs of an online auction system. This project is also aimed at creating an e-commerce environment accessible to various users of the Web.

1.4.2 Relevance

This project is relevant to the field of e-commerce. It also involves development of a web-based information system. Thus the system shall benefit both end users and administrators.

1.4.3 Significance

Successful implementation of this system will surely provide web users an alternative to buying and selling on the Internet. This is achieved from a system that handles efficiently, effectively and correctly according to the system's specifications.

1.5 System Development Approach

System development can be done in many different ways. In virtually all development projects, some variation of system development life cycle is used, with phases for project planning, analysis, design, implementation and support. This project will take the object-oriented approach where the Unified Modeling Language (UML) will be used to model and develop the system.

The object-oriented approach views an information system as a collection of interacting objects that work together to accomplish tasks. There are no processes or programs; there are no data entities or files. The system consists of objects. An object is a thing in the computer system that can respond to messages. This is a radically different view of a computer system, which requires a different approach to doing system analysis, system design and programming.

There are several key benefits of the object-oriented approach, among them are due to closeness to real-world phenomena and reusability. The object-oriented approach is natural or intuitive for people because people tend to think about the world in terms of tangible objects. Also, because the object-oriented approach involves classes of objects, and many other systems may use the same objects, these classes can be used over and over again whenever they are needed. There is less need to "reinvent the wheel" to create an object.

1.6 Methodology

A system development methodology is a very formal and precise system development process that defines a set of activities, methods, best practices, deliverables and automated tools to develop and maintain a system.

The methodology to be used in this project will be based on iterative and incremental development. In this methodology, the Unified Modeling Language (UML) will be used to build models as a number of small iterations rather than trying to define all the details of a model or diagram at once. The development is done in a sequence of steps, whereby each iteration adds some new information or detail. Then each iteration is evaluated, either on paper or in a working prototype, and used to produce input for the next iteration. One iteration or set of iterations can be treated as an increment of the system. An increment is sometimes called a version of the system and the development is based on the definition of a number of steps. After each step, actions can be taken to

correct any problems; and the experiences from each step will be input to the next step. Every step should add new functionality or attributes to the system.

Each iteration in the methodology will basically include these phases:

- *Analysis* – To identify the overall data, processes and interfaces requirements of the system. Coming up with the best combination of technical, operational, and risk feasibility.
- *Design* – To design the specifications for construction.
- *Implementation* – Constructing the system according to design specifications. Loading files and databases for testing on the system.
- *Testing* – To test the capability and quality of the system.

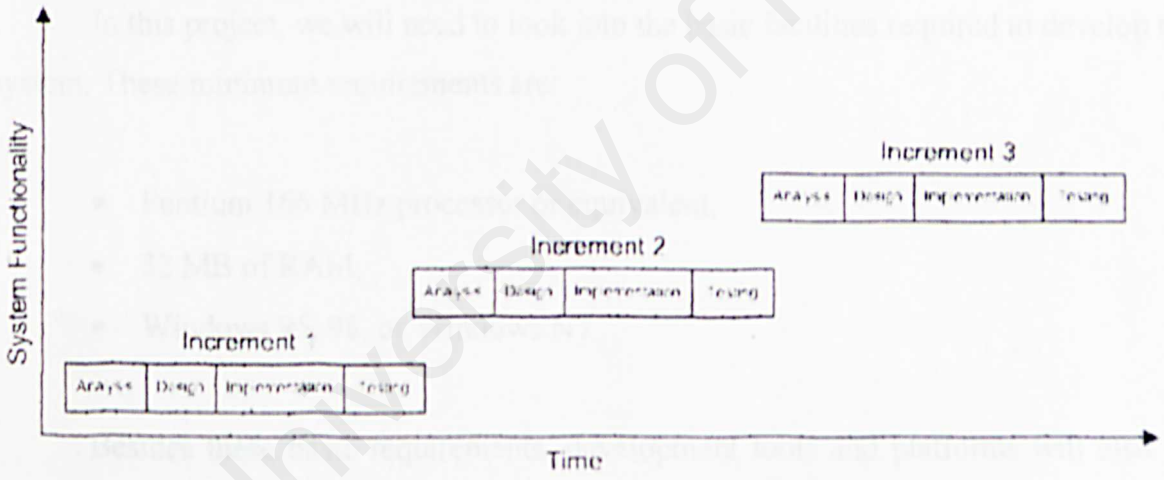


Figure 1.1 Iterative and Incremental Development

1.7 Project Schedule

ID	Task Name	Start	Finish
1	Preliminary Investigation	Mon 6/4/01	Sun 7/22/01
2	Analysis	Mon 7/9/01	Mon 10/1/01
3	Design	Tue 7/24/01	Sun 10/14/01
4	Construction	Wed 8/8/01	Sun 10/28/01
5	Implementation & Testing	Thu 8/23/01	Sun 11/11/01
6	Evaluation	Mon 11/12/01	Sun 11/25/01

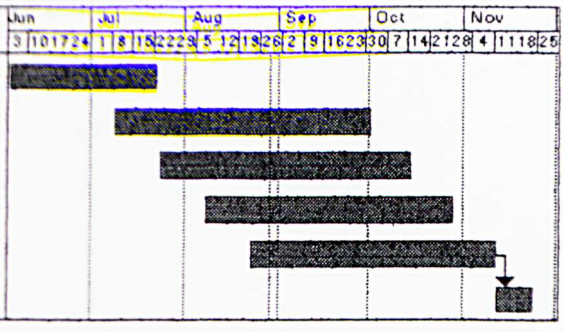


Figure 1.2 Project Schedule

1.8 Facilities Required

In this project, we will need to look into the basic facilities required to develop the system. These minimum requirements are:

- Pentium 166 MHz processor or equivalent,
- 32 MB of RAM,
- Windows 95, 98, or Windows NT

Besides these basic requirements, development tools and platforms will also be needed to develop the system. These aspects shall be looked into in the analysis phase where various considerations are made.

1.9 Expected Outcomes

For this project, it is expected that a fully functional web based system is developed. This system shall include the following components:

- Registration and login system
- Item bidding and posting system
- User information management
- Item management
- User rating system
- Maintenance system

Besides that, supporting documents including the user manuals and administrator manual will also be prepared. All these components and documentation will be compiled and a report shall be submitted before the project dateline.

CHAPTER 2 : LITERATURE REVIEW

The purpose of this literature review is to gather as much relevant information as possible about the Virtual Auctions project. It is very important as it recognizes other projects that might have similar characteristics to this project. Therefore, a literature review will be able to help the developer know the features offered by existing systems.

Besides that, the developer can rather concentrate on learning the important aspects of an existing system than to reinvent the wheel that has already been invented. The literature review can also equip the developer with the various development tools available before making further considerations in the analysis stage.

2.1 Introduction to Auctions

Auction is actually a commerce tool used mainly where the price of goods or service cannot be prejudged. It is a market-driven sales instrument unlike the seller-driven one that is common for supermarkets.

2.1.1 Classification of Auctions

There are several auction models currently practised in the marketplace for valuation of goods between buyers and sellers. The auctions can be grouped in several categories depending on the bid (submitted by the buyer) or offer (submitted by the seller), dynamic nature of bid, etc.

- *The English Auction* – The auctioneer sequentially increases the offer price to a set of buyers. The auction is stopped when only one active bidder remains or there is no response for a long time. The auctioned item is sold to the bidder with the highest bid.

- *The Dutch Auction* – The auctioneer lowers the price sequentially until the buyer agrees to pay the offer price by shouting "mine" or pressing a button. Traditionally, the prices are indicated on the dial of a clock. Instead of increasing, the clock hand falls over a price scale. The fall of price continues until a buyer presses a button to stop the clock. He who presses the button wins the auction at the current price.
- *Bid Auction* – Buyers can normally make bids sequentially but sellers can only indicate that a bid is accepted. In theory, all bidding occurs at the same moment, but in practice, it takes several minutes to complete the cycle. The winner is the bidder with the highest bid.
- *Offer Auction* – An institution in which sellers can make offers sequentially, and buyers are able to accept any offer, but not to make any bid. This allows the customer to choose or not to choose the offer by the supplier.
- *Seller Posted-Offer Auction* – Sellers independently select a price and a maximum quantity limit at the beginning of each trading period. The trading period ends when all buyers have had an opportunity to shop or when all sellers are out of stock.
- *Double Auction* – The bids, offers and trading prices are public information. Both buyers and sellers are free to enter price quotes into the market. Under the rules of the double auction, any buyer who makes a bid, that bid is publicly announced to the market. Sellers' offers are also publicly announced. This pricing institution characterizes the floor trading on many stock and commodity exchanges throughout the world.
- *Posted-Bid Auction* – Buyers submit their bids and these are publicly displayed. The sellers approach in random order to make sales decisions. This situation is the reverse of the posted-offer auction.
- *Clearing-House Auction* – Buyers submit bids and sellers submit offers. The submitted bids are displayed in descending order and offers are displayed in ascending order. The price of the auction item is selected by crossing the two lists.
- *Discriminative Auction* – One or more identical items are offered for sale at the same time. Buyers submit posted bids to a single seller, who offers a fixed

number of units, N, to the bidders whose bids are among the N highest bids at their price.

- *First-Price Sealed-Bid Auction* – A discriminative auction but with only one unit or "prize" to be auctioned. The government and large corporate bodies commonly deploy this auction.
- *Competitive Sealed-Bid Auction* – In contrast to the discriminative case, it is possible to design a mechanism for selling multiple units in which all bidders whose bids are among the N highest (winning) bids pay a uniform price. The uniform price is specified to the highest rejected bid.
- *Second-Price Sealed-Bid Auction* – When the bids are opened, the bidder with the highest bid pays the amount of the second highest bid and wins the single object.
- *Decentralized Negotiation* – Each seller (or buyer) is allowed to roam freely around the room and negotiate contracts. After the contract is completed, the buyer and seller report to the central point and the price is usually on the blackboard at the time it is reported.

2.1.2 Auction Protocol

All auctions practised must be guided by some sort of protocol. The non-electronic auctions, such as English and Dutch auctions, use physical norms understandable by all parties involved. In the electronic version of auctions, the bidders no longer need to be confined near a stage. The service normally follows the following steps:

1. Registration of buyers and bidders using secure transaction.
2. Search for auctioned items in the site.
3. Placing bids and processing.
4. Winner declaration and deal processing.
5. Transfer of auction item and collection of bid.

2.1.3 Major Challenges of Online Auctioning

The major barrier for the online auction is the payment and goods transportation. The Internet environment allows the auction environment to be changed but at the same time introduces new problems of its own. Here are some of the constraints faced:

- *Security and Trust Issues* – The auction needs the trust that the auctioned item will be shipped as described in the auction list. At the same time, the seller needs the assurance that when the item is delivered, the buyer will pay the amount due. The information regarding the personal details passed to the auction house should not be misused.
- *Unlawful Conduct Issues* – Rules and regulations regarding auctions over the Internet vary from state to state and country to country. How this should be maintained is a nightmare for auction houses. The frauds in auction also need to be framed in order to deal with the law.
- *Bidder Collusion* – A set of bidders may collude and form a ring, where members of the ring agree not to outbid each other. At the end of the auction, if a member of the ring owns the item, it is resold among the other members in a separate auction or some other allocation procedure.
- *Choice of Right Type of Auction* – Several types of auctions are available on the Internet. The choice of auction is important for the success of the auction. The governments are most likely to lean towards auction methods that have higher transparency.
- *Withdrawing Options for English Auctions* – Online auctions normally lasts for hours or even days and some bidders may hesitate to commit for such a long time. As such, the bidder must be given a chance to withdraw his/her bid. This also means that there should be some kind of mechanism not to allow re-entry in auction at a later time.

2.2 Background Survey

In this part of the Literature Review, we will look into a popular web-based auction house currently available in the Internet.

2.2.1 eBay ™

A leading company in the online auctions sector is eBay. It is one of the most profitable e-commerce methods around. Linda and Pierre Omidyar started this successful online auction site in 1995. The site was called AuctionWeb at first but later renamed eBay and has since become the premier online auction house. It has as many as 4 million unique auctions in progress and 450,000 new items added each day [DEI99].

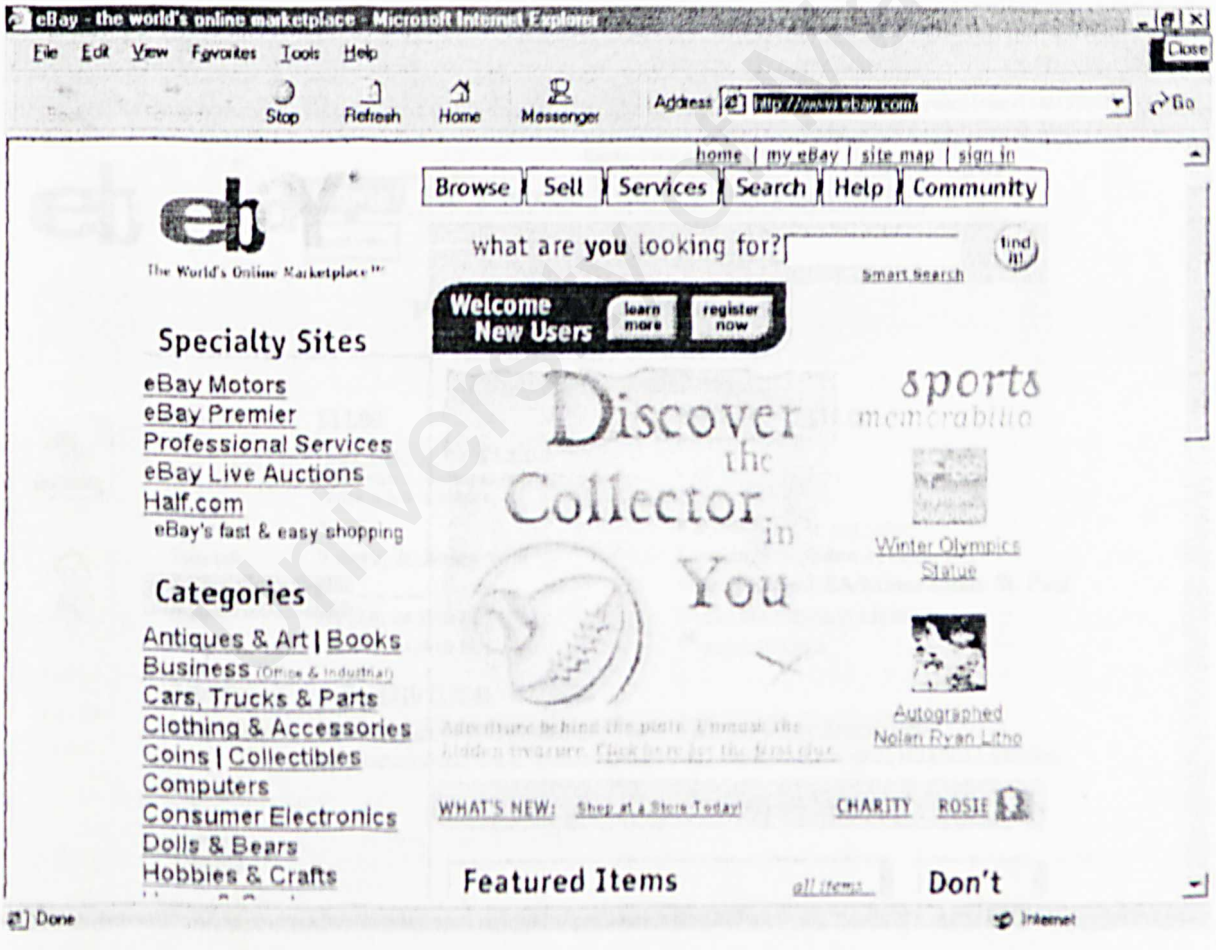


Figure 2.1 The eBay home page

On eBay, people can buy and sell almost anything we can think about. The company collects a submission fee plus a percentage of the sale amount. The submission fee is based on the amount of exposure sellers want their items to receive. This means that a seller will have to pay more if he or she wants to post the item among the "featured auctions" category. Even listing the product in a bold-face font requires a higher submission fee!

A database is used by eBay to manage the millions of auctions that it offers. This database does not only manage information about auctions but also the user's personal information and other relevant information. Therefore, when a potential bidder sees the item he or she is interested in, he or she can also view the profile of the seller.

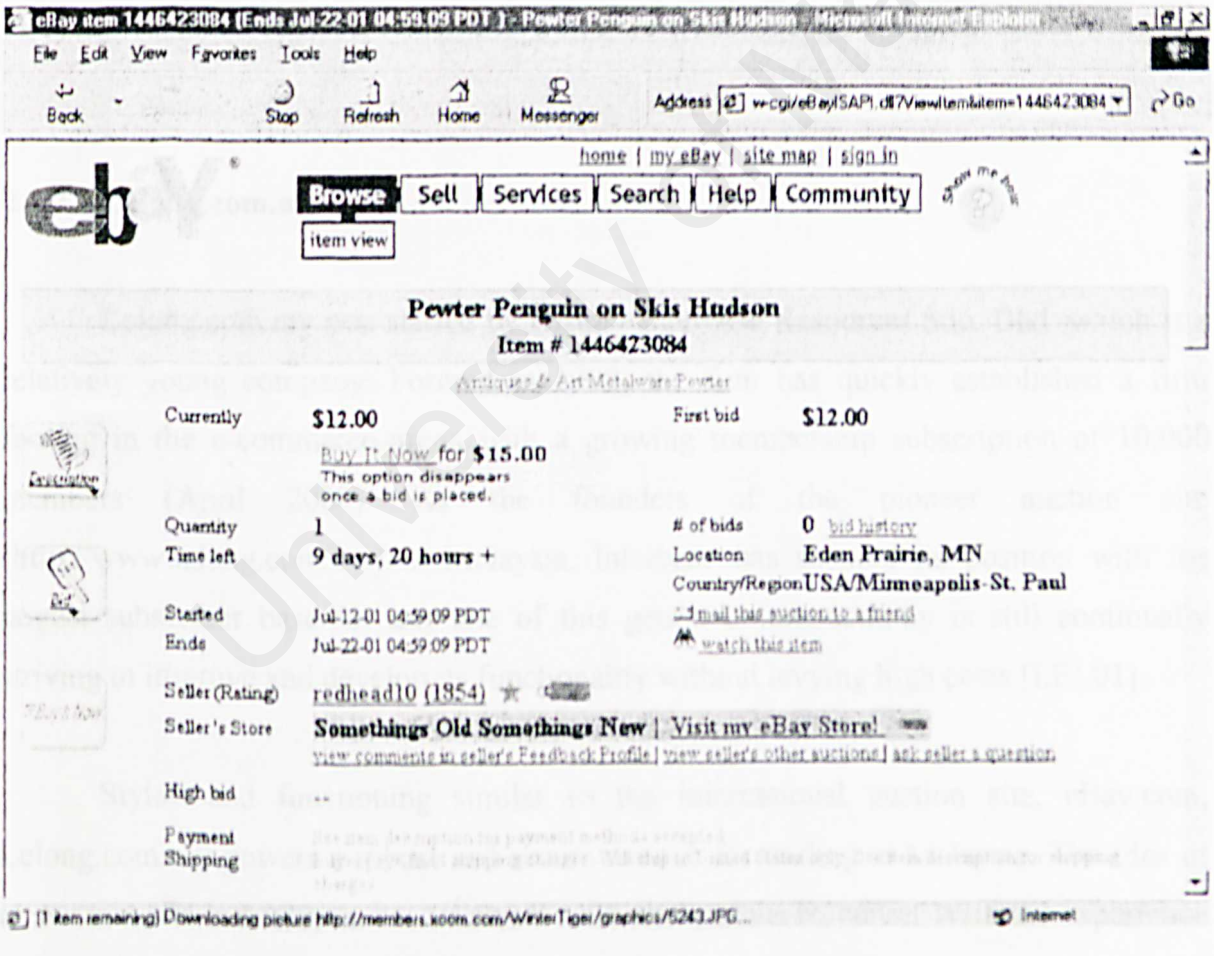


Figure 2.2 Placing a bid on eBay

The auction process begins when the seller pests a description of the item for sale and fills in the appropriate registration information. The seller must specify a minimum opening bid. If potential buyers feel this price is too high, the item may not receive any bids. In many cases, a reserve price is also set. Sellers might set the opening bid lower than the reserve price to generate bidding activity. If a successful bid is made, the seller and buyer negotiate the shipping details, warranty and other particulars.

A number of new businesses now use eBay as their means of selling products. These businesses depend on eBay to remain up and running. Therefore, eBay invests in high-availability computing and continuous-availability computing to prevent or minimize downtime [DEI99].

By implementing traditional marketing strategies and keeping the process simple, eBay has offered a successful alternative to store-front style e-commerce.

2.2.2 Lelong.com.my

Lelong.com.my was started by History Interbase Resources Sdn. Bhd. which is a relatively young company. Formed in 1998, the firm has quickly established a firm footing in the e-commerce arena with a growing membership subscription of 10,000 members (April 2000). As the founders of the pioneer auction site (http://www.lelong.com.my) in Malaysia, Interbase has secured its position with the largest subscriber base for any site of this genre. Lelong.com.my is still continually striving to improve and develop its functionality without levying high costs [LEL01].

Styled and functioning similar to the international auction site, eBay.com, Lelong.com.my powers a robust and secure avenue for trading in Malaysia. The idea of an auction site in Malaysia has taken off with tremendous response. With the experience gained from running the site for the past years, Interbase is well poised to deliver a site that is full of quality content to its members. Starting based on Consumer-to-Consumer

(C2C) transactions, Lelong.com.my is now moving to becoming a channel for Business-to-Consumer (B2C) relationships. Many distributors have realized the power of using the auction system and are now using Lelong.com.my to trade and grow their businesses.

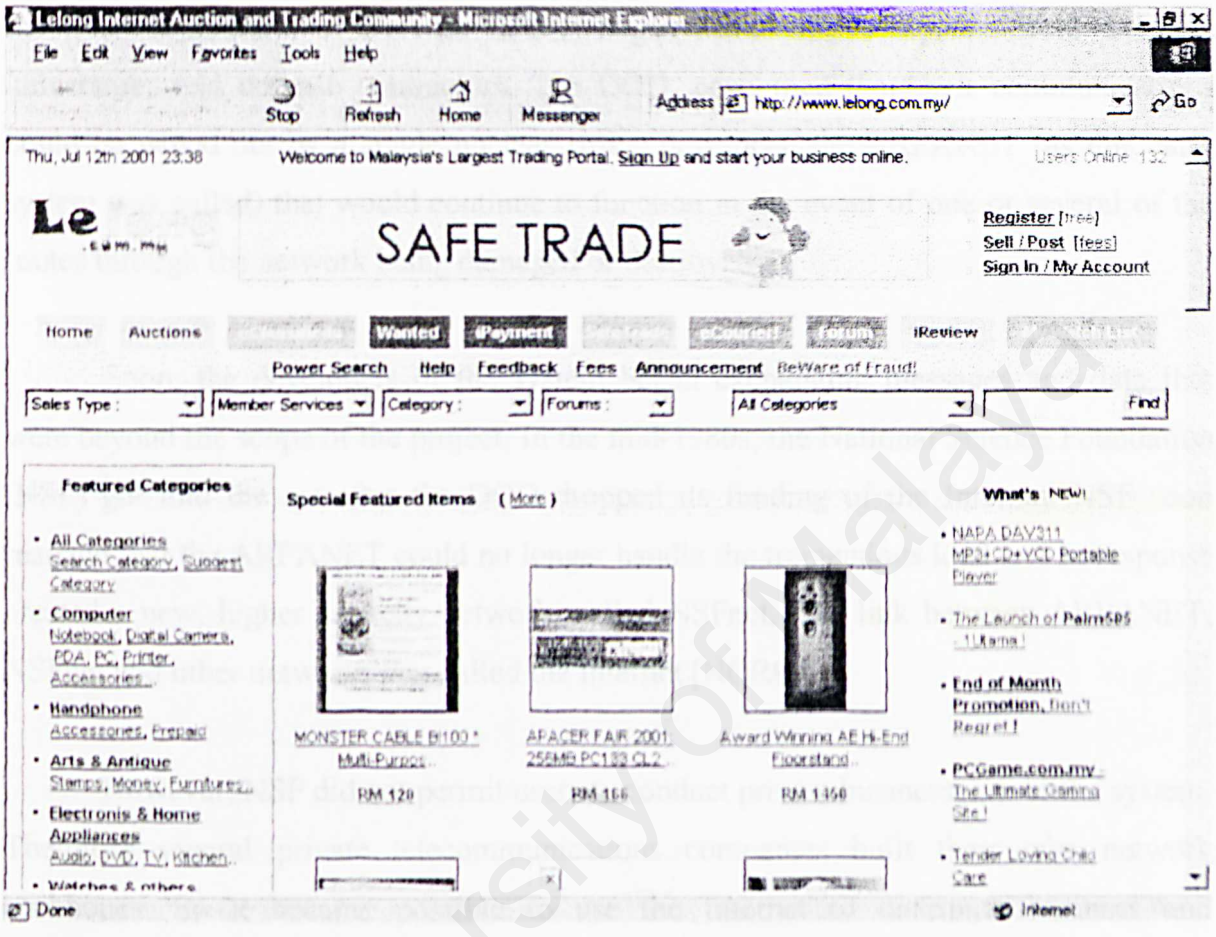


Figure 2.3 The Lelong.com.my home page

2.3 Introduction to the Internet

The Internet can be defined as a large network of computers. The first seeds of the Internet were planted in 1969 when the Advanced Research Projects Agency (ARPA) of the U.S. Department of Defense (DOD) began connecting computers from different universities and defense contractors. The DOD, concerned that their communications could be wiped out by a single nuclear strike, developed the ARPANET (as this early system was called) that would continue to function in the event of one or several of the routes through the network being damaged or destroyed.

Soon, the developers of the system began exchanging messages and data that were beyond the scope of the project. In the mid-1980s, the National Science Foundation (NSF) got into the act after the DOD dropped its funding of the Internet. NSF soon realized that the ARPANET could no longer handle the tremendous load and in response created a new, higher capacity network, called NSFnet. The link between ARPANET, NSFnet and other networks was called the Internet [NOR99].

However, NSF did not permit users to conduct private businesses over the system. Therefore several private telecommunications companies built their own network backbones. So it became possible to use the Internet to distribute business and commercial information.

The original ARPANET was shut down in 1990, and the government funding for NSFnet was discontinued in 1995, but the commercial Internet backbone services have easily replaced them. By the early 1990s, interest in the Internet expanded dramatically. Now, the Internet has not only become an information superhighway but also a virtual community and marketplace for all those who have access to it.

2.4 Client/Server Computing

2.4.1 Definition of Client/Server Computing

Client/server is a computational architecture that involves client processes requesting service from server processes. It is the logical extension of modular programming. Modular programming has as its fundamental assumption that separation of a large piece of software into its constituent parts ("modules") creates the possibility for easier development and better maintainability [DIX96]. Client/server computing takes this step farther by recognizing that those modules need not be executed within the same memory space. With this architecture, the calling module becomes the "client" (that which requests a service), and the called module becomes the "server" (that which provides the service).

The client is a program that sends a message to a server program, requesting that the server perform a task. Client programs usually manage the user-interface portion of the application, validate data entered by the user, dispatch requests to server programs, and sometimes execute business logic. The client-based process is the front-end of the application that the user sees and interacts with. The client process contains solution-specific logic and provides the interface between the user and the rest of the application system.

A server program fulfills the client request by performing the task requested. Server programs generally receive requests from client programs, execute database retrieval and updates, manage data integrity and dispatch responses to client requests. Sometimes server programs execute common or complex business logic. The server-based process "may" run on another machine on the network. This server could be the host operating system or network file server; the server is then provided both file system services and application services [DIX96].

2.4.2 Two-Tier Architecture

A two-tier architecture is where a client talks directly to a server, with no intervening server. It is typically used in small environments (less than 50 users). A common error in client/server development is to prototype an application in a small, two-tier environment and then scale up by simply adding more users to the server. This approach will usually result in an ineffective system, as the server becomes overwhelmed. To properly scale to hundreds or thousands of users, it is usually necessary to move to a three-tier architecture.

2.4.3 Three-Tier Architecture

A three-tier architecture introduces a server (or an "agent") between the client and the server. The role of the agent is manifold. It can provide:

- Translation services - as in adapting a legacy application on a mainframe to a client/server environment,
- Metering services - as in acting as a transaction monitor to limit the number of simultaneous requests to a given server, or
- Intelligent agent services - as in mapping a request to a number of different servers, collating the results, and returning a single response to the client.

2.5 Web Programming

In this part of the literature review, we will look at some of the latest technologies and tools currently used in web developments.

2.5.1 ASP

Active Server Pages (ASP) is an open, compile-free application environment in which we can combine HTML, scripts, and reusable ActiveX server components to create dynamic and powerful Web-based business solutions. Active Server Pages enables server side scripting for IIS with native support for both VBScript and JScript.

Files created with Active Server Pages have a ".asp" extension. With ASP files, web sites can be activated using any combination of HTML, scripting-such as JavaScript or Visual Basic® Scripting Edition (VBScript) - and components written in any language [RES00]. This means an ASP file is simply a file that can contain any combination of HTML, scripting, and calls to components. When a change is made on the ASP file on the server, the administrator needs only to save the changes to the file - the next time the Web page is loaded, the script will automatically be compiled. It works because ASP technology is built directly into Microsoft Web servers, and is thus supported on all Microsoft Web servers: Windows NT Internet Information Server (IIS) 3.0, Windows NT Workstation, and Windows 95 Personal Web Server.

When a browser requests an ASP file from a Web server, the Web server calls Active Server Pages to read through the ASP file, executing any of the commands contained within and sending the resulting HTML page to the browser. An ASP file can contain any combination of HTML, script, or commands. The script can assign values to variables, request information from the server, or combine any set of commands into procedures [RES00].

The scripting languages supported by ASP in turn support use of the If-Then-Else constructs. Therefore, ASP enables embodiment of some real logic into HTML. ASP uses the delimiters, "<%" and "%>" to enclose script commands. For example, the code below sets the value of the variable "Color" in the user cookie to "blue":

```
<% Response.Cookies("Color") = "blue" %>
```

ASP includes five standard objects for global use:

- *Request* - to get information from the user
- *Response* - to send information to the user
- *Server* - to control the Internet Information Server
- *Session* - to store information about and change settings for the user's current Web-server session
- *Application* - to share application-level information and control settings for the lifetime of the application

The *Request* and *Response* objects contain collections (bits of information that are accessed in the same way). Objects use methods to do some type of procedure and properties to store any of the object's attributes (such as color, font, or size).

2.5.2 CGI

The Common Gateway Interface (CGI) is a standard for interfacing external applications with information servers, such as HTTP or Web servers. A plain HTML document that the Web browser retrieves is static, which means it exists in a constant state: a text file that does not change. A CGI program, on the other hand, is executed in real-time, so that it can output dynamic information [DEI99].

For example, if we wanted to "hook up" a Unix database to the World Wide Web, to allow people from all over the world to query it. Basically, we need to create a CGI

program that the Web browser will execute to transmit information to the database engine, and receive the results back again and display them to the client. This is an example of a gateway, and this is where CGI, currently version 1.1, got its origins.

Since a CGI program is executable, it is basically the equivalent of letting the world run a program on our system, which is not the safest thing to do. Therefore, there are some security precautions that need to be implemented when it comes to using CGI programs. Probably the one that will affect the typical Web user the most is the fact that CGI programs need to reside in a special directory, so that the Web server knows to execute the program rather than just display it to the browser. This directory is usually under direct control of the webmaster, prohibiting the average user from creating CGI programs.

A version of the NCSA HTTP server distribution will have a directory called /cgi-bin. This is the special directory mentioned above where all of CGI programs currently reside. A CGI program can be written in any language that allows it to be executed on the system, such as:

- C/C++
- Fortran
- PERL
- TCL
- Any Unix shell
- Visual Basic

It just depends what is available on the system. If a programming language like C or Fortran is used, the program must be compiled before it will run. If, however, one of the scripting languages is used instead, such as PERL, TCL, or a Unix shell, the script itself only needs to reside in the /cgi-bin directory, since there is no associated source code [DEI99]. Many people prefer to write CGI scripts instead of programs, since they are easier to debug, modify, and maintain than a typical compiled program.

2.5.3 HTML

The Hypertext Markup Language (HTML) is the lingua franca of the Net. It is a simple, universal mark-up language that allows Web publishers to create complex pages of text and images that can be viewed by anyone else on the Web, regardless of what kind of computer or browser is being used.

There is no need for special software to create an HTML page; all that is needed is a word processor (such as Microsoft Notepad, or Microsoft Word) and a working knowledge of HTML. Basic HTML is indeed very easy to learn.

HTML is just a series of tags that are integrated into a text document. HTML is like stage directions - silently telling the browser what to do, and what props to use.

HTML tags are usually English words (such as blockquote) or abbreviations (such as "p" for paragraph), but they are distinguished from the regular text because they are placed in small angle brackets. So the paragraph tag is <p>, and the blockquote tag is <blockquote>. Some tags dictate how the page will be formatted (for instance, <p> begins a new paragraph), and others dictate how the words appear (<b> makes text bold). Still others provide information - such as the title - that does not appear on the page itself.

The first thing to remember about tags is that they travel in pairs. Every time a tag is used - say <blockquote> - it must always be closed off with another tag - in this case, </blockquote>. The slash - / - before the word "blockquote" is what distinguishes a closing tag from an opening tag.

The basic HTML page begins with the tag <html> and ends with </html>. In between, the file has two sections - the header and the body. The header - enclosed by the <head> and </head> tags - contains information about a page that will not appear on the page itself, such as the title. The body - enclosed by <body> and </body> - is where the action is. Every thing that appears on the page is contained within these tags.

2.6 Web Server Operating Systems

2.6.1 Microsoft Windows NT Server

Since it was first released in 1993, Microsoft Windows NT Server has established itself as the network operating system (NOS) of choice for countless organizations of all sizes in private industry and public agencies. They have discovered that Windows NT Server is extremely reliable, highly scalable, and capable of handling the complex, mission-critical demands.

Windows NT Server stems from more than just its operating system capabilities. It provides a complete and solid platform for an all-encompassing range of services and activities. It combines the best aspects of an application server, a file and printer server, a communications server, and a Web server - along with interoperability and management features that make it an excellent NOS for organizations, whether they have mixed computing environments or operate entirely on Windows NT Server [MIC01].

Windows NT Server provides the backbone for a complete, organic system, where all elements working together seamlessly. When joined with other Windows NT-related products, including the BackOffice family of applications and Windows NT Workstation, Windows NT Server provides the foundation for a powerful and well-integrated environment.

Windows NT Server is an operating system that includes powerful, comprehensive Web platform - Internet Information Server (IIS). IIS technology enables the creation of powerful Web servers instantly, because all of the capabilities are native to the operating system. Instead of patching together separate components of a Web platform, IT managers using Windows NT Server with IIS have all the components necessary for establishing Internet services, including Web, FTP, and Gopher.

The same security features used for Windows NT Server, which gives any Web server robust protection without any extra work required by administrators. These include Secure Sockets Layer 3.0 and Microsoft Certificate Server, a general-purpose, customizable server application for managing digital certificates.

Fast Web server performance, due to the fact that Web services are optimized through a single-process design with multithreaded architecture.

Windows NT Server has built-in Web page design and scripting features including the FrontPage Web authoring application and Active Server Pages (ASP). ASP allows HTML authors and Web developers to intermix HTML with in-line scripting using almost any authoring tool. Active Server Pages include native support for Visual Basic Scripting Language and JScript development software, and are compatible with any ActiveX scripting language [MIC01].

2.6.2 UNIX

The UNIX operating system was developed at AT&T Bell Laboratories in Murray Hill, New Jersey, USA. Since the original design and implementation of the UNIX system by Ken Thompson in 1969, it has gone through a maturing process [SOB95]. When the UNIX operating system was developed, many computers still ran single jobs in a batch mode. Because these systems served only one user at a time, they did not take full advantage of the power and speed of the computers.

The UNIX time-sharing system provided three major improvements over single-user, batch systems:

- It allowed more than one person to use the computer at the same time,
- It allowed a person to communicate directly with the computer via a terminal, and
- It made it easy for people to share data and programs.

When the UNIX operating system became widely available in 1975, Bell Labs offered it to educational institutions at a minimal cost. As these students graduated and went into the industry, they expected to work in a similarly advanced environment. As more of these students worked their way up in the commercial world, the UNIX operating system found its way into the industry.

In addition to introducing its students to the UNIX operating system, the Computer System Research Group at the University of California at Berkeley made significant changes and additions to it. They made so many popular changes that one of the two most prominent versions of the system in use today is called the Berkeley Software Distribution (BSD) of the UNIX system. The other major version is the UNIX system V, which descended from versions developed and maintained by the AT&T and UNIX System Laboratories [SOB95].

The UNIX is a generic operating system and it takes advantage of available hardware power. Because the UNIX system was written almost entirely in a machine-independent language, it can be easily adapted to meet special requirements. Equally important, it was originally designed as a multi-user operating system – it was not modified to serve several users as an afterthought.

The largest computer market today is for personal computers, most of which run on MS-DOS or Windows. PCs were meant to be used by a single user at one time, so the multi-user, multitasking benefits of the UNIX system were of little interest in the PC community. The UNIX system, however, is typically the operating system of choice for many workstations and many larger systems.

CHAPTER 3 : ANALYSIS

In this phase of the project, we will look into the systems analysis phase. Systems analysis is a problem-solving technique that decomposes a system into component pieces for the purpose of studying how well those component parts work and interact to accomplish their purpose. For this project, we will be using the Object Oriented Analysis (OOA) method in the analysis phase. OOA is a model-driven technique that integrates data and process concerns into constructs called objects. OOA models are pictures that illustrate the system's objects from various perspectives such as structure and behavior.

3.1 Requirements Analysis

In this phase of the project, we are going to define the requirements of the Virtual Auctions System. The requirements analysis may seem to be an easy or trivial task but it is often the source of many errors, omissions and conflicts. Therefore, this phase shall be carefully done by translating the objectives of this project into an outline of functional and nonfunctional requirements that will be needed to meet the objectives.

3.1.1 Functional Requirements

A functional requirement is a description of activities and services a system must provide. These requirements are frequently identified in terms of inputs, outputs, processes and stored date that are needed to satisfy the system's objectives.

Here are the functional requirements of the Virtual Auctions System:

i. *User Registration and Login* – The system shall provide a service for users to register as members of the auction site and to login. The registration process gets

the user's information such as name, e-mail address, etc. This service is important to identify users that use the services of the system. Therefore a proper and safe authentication system is needed to satisfy this requirement. The login service also serves as the entrance for administrators to perform certain administration tasks.

ii. *User Information and Password Management* – The service shall provide a service for members to change their personal information. This service provides the flexibility for users to update their information on the system without going through the administrator. For increased security, the system shall allow users to change their password whenever they want.

iii. *Search and Browsing Facility* – The system shall allow users to search and browse for items in the auction site. The search facility shall enable users to type in certain keywords and in return the system provides relevant search results. A proper method shall also be developed to enable users to browse items in the auction site systematically.

iv. *Bid Placement and Item Posting* – The system must be able to provide a service for members to post or bid for items on the auction site. Posting an item requires the member to give relevant details about the item to be posted while bidding for an item requires the member to specify and confirm the bidding amount. This service is only made available for registered and logged in members of the auction site.

v. *User Rating* – In order to curb the misuse of the auction site, the system shall provide a service where other members of the site can identify non-genuine buyers or sellers. This can be done through a user rating service to be provided by the system. This service allows other users to rate or comment on a particular buyer or seller after a transaction has been made. Therefore, genuine buyers or

sellers can be acknowledged while non-genuine users will eventually be eliminated by it's poor rating.

vi. *Maintenance* – The system must provide a service for administrators of the system to perform certain administration tasks on the system. These services shall include:

- *Category Management* – Administrators can add or delete a category in the system depending on the current needs of the auction site.
- *User Management* – Allows administrators to view user information and rating; to determine users provide consistent information. Administrators can also delete or cancel memberships if that particular buyer or seller is found non-genuine through complaints.
- *Report Generation* – Administrators can perform queries on the system's database to extract certain information needed for reporting.

3.1.2 Nonfunctional Requirements

Nonfunctional requirements are descriptions of features, characteristics and attributes of the system as well as any constraints that may limit the boundaries of the proposed solution. There are mainly classifications of nonfunctional requirements; mainly performance, information, economy, control, efficiency and service.

The nonfunctional requirements of the Virtual Auctions System are:

i. *Ease of Learning* – The auction services of the system shall be easily learned by users. This requirement is very important because users of the system may come from various cultures and backgrounds. Thus it is indeed a challenge to create system that is easy for people of all walks of life to learn how to bid and sell at the auction site.

ii. *User Friendliness* – The interface provided by the Virtual Auctions system shall be user friendly enough to cater for the various users. This is particularly important as convenience play a big role in ensuring the users keep coming back to the site. Therefore a user-friendly system shall also be a nonfunctional requirement of the system.

iii. *Efficiency* – Efficiency requirements represent the system's ability to produce outputs with minimal waste. Therefore duplicated steps in the system's processes must be eliminated. This can also help in increasing the performance of the system through high throughput rates and fast response times. Efficiency also means reducing wastes in the way the system uses its resources.

3.2 Functional Description of the Virtual Auctions System

The approach commonly used to model the functional aspects of the system is called use case modeling. This method identifies and describes the system's functions from the perspective of external users using a tool called use cases.

3.2.1 Actors and Use Cases

The user may be a person or another machine and is called the actor in the system. Each use case is a different way to use the system and the completion of each use case produces a different result. In our system there are two main actors:

- *User* – Represents the users that use the Virtual Auctions system to browse for items, bid for an item, etc.
- *Administrator* – The people who maintain the system and perform the administration tasks on the system.

And these are the different use cases involved:

- *Register* – A user registers as a member of the auction site to access services such as bidding and selling. It involves users giving personal information such as e-mail address, street address, etc.
- *Login* – A registered user of the auction site will have to login each time he or she wants to sell or bid for an item in the site.
- *Change Registration Info* – A user can opt to change the personal details given during registration whenever he or she wants to.
- *Change Password* – As a safety feature, users can change his or her password as often as possible through this service provided by the system.
- *Select Category* – A user can browse through the categories available in the auction site to look for items of interest located at a particular category.
- *Search for Item* – A user can type in a keyword of the item he or she is interested in and in return the system will display all the relevant items available.
- *View Item Info* – Each item posted in the auction site has some details entered by it seller. A potential buyer can view this information before deciding whether to bid for that item or not.
- *Bid for Item* – Once a user makes the decision to bid for an item, the system will provide the necessary interface for the use to make a bid by entering the specific amount of the bid.
- *Post Item* – A user can also post an item for sale by setting a minimum price for that item and determining the increment price for bidding.
- *Rate a User* – After a transaction has been made, a user can rate the seller or buyer according to the user's opinion. This rating method will look into the delivery, payment and accuracy of posting of the seller or buyer.
- *View User Rating* – Before deciding to sell or bid for an item, users can view the buyer or seller's rating to ensure genuine transactions.
- *Maintenance* – The administrator can perform administration tasks in the system through the maintenance service provided by the system.

- *Add a Category* – The administrator can add a category into the system when there is a need for a new category.
- *Delete a Category* - The administrator can delete a category off the system when there is a no need for that category due to poor response or irrelevance.
- *View User Info* – The administrator can view all the personal data provided by the user to check ensure only genuine users use the auction site.
- *Delete a User* - The administrator can delete a user off the system once that user is found to be non-genuine through poor ratings and complaints made by other users to the administrator.
- *View Report* – The administrator can also perform certain queries to the system's database to retrieve information for reporting purposes.

3.2.2 Use Case Model Diagram

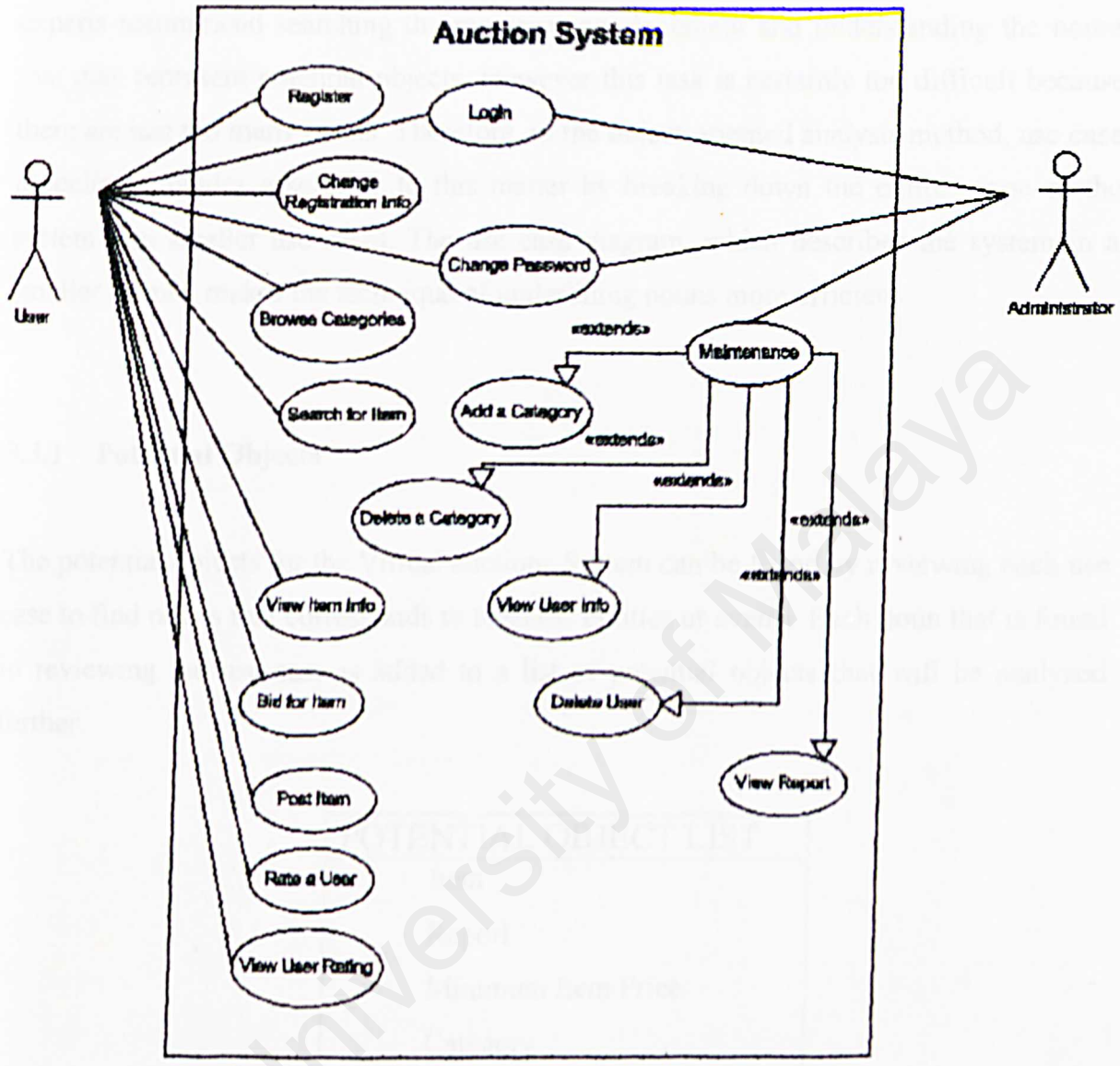


Figure 3.1 Virtual Auctions Use Case Diagram

3.3 Identification of Objects

In the process of finding and identifying objects for a system, many methodology experts recommend searching the requirements document and understanding the nouns that may represent potential objects. However this task is certainly too difficult because there are just too many nouns. Therefore, in the object-oriented analysis method, use case modeling provides a solution to this matter by breaking down the entire scope of the system into smaller use cases. The use case diagram, which describes the system in a smaller format, makes the technique of underlining nouns more efficient.

3.3.1 Potential Objects

The potential objects for the Virtual auctions System can be found by reviewing each use case to find nouns that corresponds to business entities or events. Each noun that is found in reviewing the use case is added to a list of potential objects that will be analyzed further.

POTENTIAL OBJECT LIST
Item
Report
Minimum Item Price
Category
Bid Increment
Item Reserve Price
User
User E-mail Address
User ID
Administrator
Administrator ID
Bid

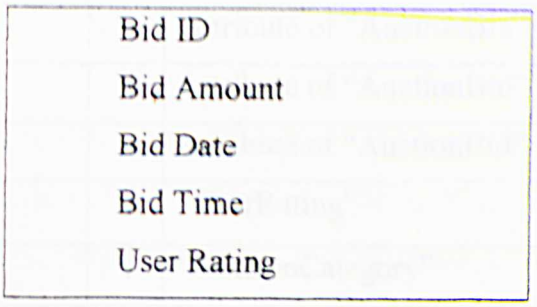


Figure 3.2 The Virtual Auctions System Potential Object List

3.3.2 Selection of Proposed Objects

Not all the candidates on the list represent genuine business objects. Now we need to "clean up" the list by removing irrelevant nouns such as synonyms, nouns outside the scope of the system and nouns that are really actions or attributes.

Figure 3.3 shows the process of "cleaning up" the list of candidate objects. A √ marks the candidates we are keeping as objects while an **X** represents the candidates that are discarded. Also listed is the explanation of why we are discarding the candidate. Finally, Figure 3.4 presents the results of the "cleaning up" process.

POTENTIAL OBJECT LIST		REASON
Item	√	"AuctionItem"
Report	**X**	Potential Interface item to be addressed in design
Minimum Item Price	**X**	Attribute of "AuctionItem"
Item Reserve Price	**X**	Attribute of "AuctionItem"
User	√	"AuctionUser"
User E-mail Address	**X**	Attribute of "AuctionUser"
User ID	**X**	Attribute of "AuctionUser"
Administrator	√	"Administrator"
Administrator ID	**X**	Attribute of "Administrator"
Bid	√	"AuctionBid"
Bid ID	**X**	Attribute of "AuctionBid"

Bid Amount	X	Attribute of "AuctionBid"
Bid Date	X	Attribute of "AuctionBid"
Bid Time	X	Attribute of "AuctionBid"
User Rating	√	"UserRating"
Category	√	"AuctionCategory"
Bid Increment	X	Attribute of "AuctionItem"

Figure 3.3 Analyzing the Potential Object List

PROPOSED OBJECT LIST
AuctionItem
AuctionCategory
AuctionUser
Administrator
AuctionBid
UserRating

Figure 3.4 The Virtual Auctions System Proposed Object List

3.4 Objects Organization and Relationships

Once the objects of the system are identified, it is time to organize those objects and document any major conceptual relationships between the objects. Finally a class diagram is used to graphically depict the objects and their associations or relationships. This diagram can also include multiplicity, associations, generalization/specialization and aggregation relationships.

3.4.1 Associations and Multiplicity

In this step we will identify associations that exist between the objects/classes. An association between two objects/classes is what one object/class "needs to know" about the other. This allows for one object/class to cross-reference another object. Once the association has been identified, the multiplicity that governs the association must be defined.

We are going to identify possible relationships by using an object/class matrix. This matrix lists the object/class as column headings as well as row headings. The name of the association and the multiplicity is recorded directly in the intersection cell of the matrix. For example, the intersection cell of row 1 and column 2 represents " Each **AuctionItem** *Belongs to 1* **AuctionCategory** ".

	AuctionItem	Auction-Category	AuctionBid	AuctionUser	UserRating	Admminis-trator
AuctionItem		*Belongs to 1*	*Has 0..**	*Belongs to 1*		
Auction-Category	*Has 0..**					
AuctionBid	*Refers to 1*			*Belongs to 1*		
AuctionUser	*Has 0..**		*Has 0..**		*Has 1*	
UserRating				*Belongs to 1*		
Admminis-trator						

Figure 3.5 Association and Multiplicity Matrix

3.4.2 Class Diagram

Figure 3.6 is a UML class diagram for the Virtual Auctions system. The model depicts the objects/classes within the domain of the Virtual Auctions system. The diagram also reflects the class associations and multiplicity that was identified earlier.

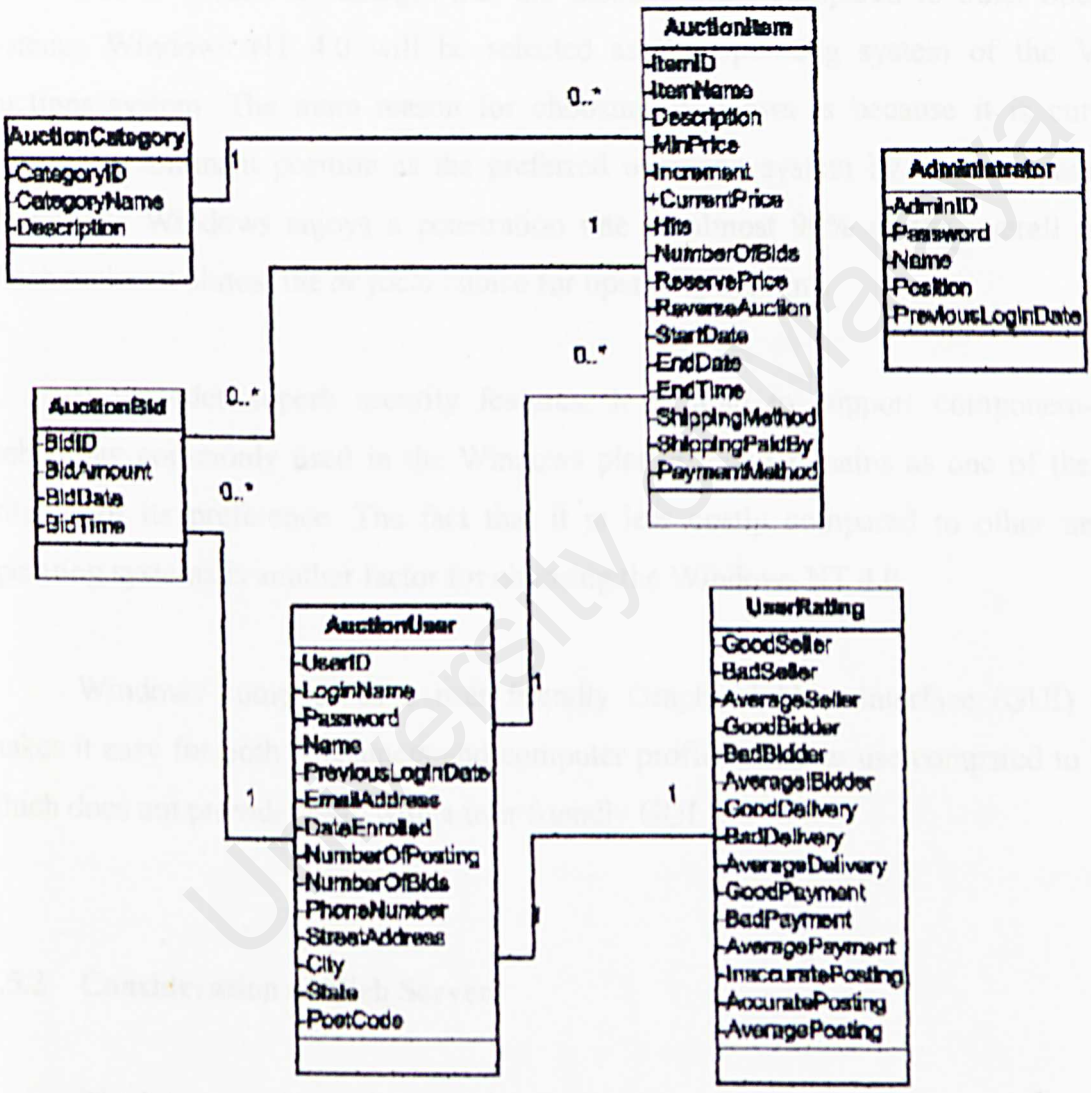


Figure 3.6 The Virtual Auctions System Class Diagram

3.5 Tools and Platforms

In this part of the project, we will determine the tools and platforms needed to develop the Virtual Auctions system.

3.5.1 Consideration of Web Operating System

Due to several advantages that are distinct when compared to other operating systems, Windows NT 4.0 will be selected as the operating system of the Virtual Auctions system. The main reason for choosing Windows is because it is currently enjoying a dominant position as the preferred operating system by most corporations. Microsoft's Windows enjoys a penetration rate of almost 90% of the overall market which makes it almost the *de facto* choice for operating systems.

It provides superb security features. It is able to support component-based technology commonly used in the Windows platform and remains as one of the main criteria for its preference. The fact that it is less costly compared to other network operating systems is another factor for choosing the Windows NT 4.0.

Windows comprise of a user friendly Graphical User Interface (GUI) which makes it easy for both consumers and computer proffesionals to use compared to UNIX which does not provide users with a user friendly GUI.

3.5.2 Consideration of Web Server

The Internet Information Server 4.0 (IIS 4.0) will be selected as the web server for this system as it comes with the Windows NT 4.0 operating system and therefore is tightly integrated with it. Moreover, it provides a comprehensive web server and web publishing system designed especially for the use with the Microsoft Windows NT 4.0 operating system.

Besides that, it also allows security features to be implemented on specific virtual directories to control access. Since ASP is to be used as the web programming technology for implementing the Virtual Auctions system, the IIS will be most suitable as ASP runs on the IIS.

3.5.3 Consideration of Web Application Language

HTML is the universal language in the Internet. It remains as the default selection for developing web pages. It's high degree of interoperatibility enables the Virtual Auctions system to be accessed virtually from anywhere in the world. It is easy to learn and use even by a novice.

Although it may be static, some web pages achieve its desired objectives of information dissemination. Thus HTML plays its role as the web spplication language to publish static information on the internet.

3.5.4 Consideration of Web Programming Technology

The Active Server Pages (ASP) will be used as the web programming technology for this project due to several reasons. It is chosen over the Common Gateway Interface (CGI) for its ability to provide improved performance. In request, ASP is able to create a HTML file to be returned to the user. Changes made to the program do not require the program to be recompiled.

ASP provides confidentiality for specific codes from the user of the Internet as code written in ASP will only be displayed as pure HTML when viewed using an Internet browser. The display of pure HTML as an output is a benefit by itself as it makes output from ASP compatible to all web browsers.

ASP pages can easily be written, which is also one of the major factors which encouraged the usage of this technology. It is a powerful technology from Microsoft which promises a bright future in the IT industry. Using the ASP in implementing the Virtual Auctions system could be an opportunity to master this technology.

3.5.5 Consideration of Web Application Development Tools

Although Microsoft's Visual Interdev 6.0 is considerably more difficult over other tools, it is still recommended as the web aplication development tool for ASP programming. It offers considerable advantages as it is very powerful and has many features which makes it exciting for web developers to use.

It is chosen over Microsoft's Frontpage due to the fact that Frontpage will 'autoedit' a page and result in the pages not being displayed as expected. Furthermore the Visual Interdev boasts strong links with the SQL server, which makes it very easy to set up databases combined with ASP.

3.5.6 Consideration of Scripting Language

There is undoubtly an inclination to use VBScript as many books and sources of information tend to recommend and promote the use of VBscript as the scripting language when coding using ASP. In addition, it is chosen as ASP is already chosen as the technology to implement the Virtual Auctions system.

As ASP processes its codes on the server side, the usage of VBScript or any other type of scripting language has no effect or whatsoever on the outcome displayed in the web browsers. Due to the preceding two facts, VBScript was chosen as the scripting language of choice.

Chapter 4 : Design

Object-oriented design is the continuation of object-oriented analysis, continuing to center the development focus on object modeling techniques. In the analysis phase, we have already identified objects and use cases based on ideal conditions and independent of any hardware or software solution. During this stage of the project, we will refine those objects and use cases to reflect the actual environment of the proposed solution.

Object-oriented design includes the following activities:

- Refining the use case model to reflect the implementation environment.
- Modeling object interactions and behavior that support the use case scenario.

4.1 Use Case Model Refinement

In this iteration of use case modeling, the use case will be refined to include details of how the user and administrator will actually interface with the system and how the auction system will respond to that stimulus. The manner in which the user accesses the system – via a menu, window or button – will be described in detail.

In the following stages we will adapt each use case to the implementation environment or "reality" and document the results. These refined use cases can later be used for programming and interface specifications.

4.1.1 Design Use Cases

In this stage, we sill refine each of those use cases modeled during systems analysis to reflect the physical aspects of the implementation environment for the Virtual Auctions system.

Use Case Name:	Register	
Actor(s):	User	
Description:	A user registers as a member of the auction site to access services such as bidding and selling. It involves users giving personal information such as e-mail address, street address, etc.	
Typical Course of Events:	**Actor Action** The registration window is displayed when the user clicks the "Register" link from the auction site. **Step 1:** The user keys in his/her particulars in the text boxes that are labeled NAME, E-MAIL ADDRESS, PHONE NUMBER, STREET ADDRESS, CITY, STATE and POSTCODE. The user also needs to key in a LOGIN NAME and PASSWORD. **Step 2:** After filling up the registration particulars, the user clicks the [Submit] button to submit the form.	**System Response** **Step 3:** The system verifies that all the text boxes are filled and that the LOGIN NAME submitted by the user is unique from all the other records of LOGIN NAME already contained in the database. **Step 4:** After the verification process, the system assigns a unique USER ID to the user and creates a new record of the object AuctionUser in the database containing all the information submitted by the user. **Step 5:** The system replaces the current window a new window confirming the registration of the user. This window also lists out all the particulars entered by the user including the LODIN NAME and PASSWORD.
Alternate Courses:	**Step 4:** If not all the text boxes have been filled or the LOGIN NAME submitted is not unique, the system displays the registration window again and includes a message notifying the user to re-submit the form. Go to **Step 1**.	

Figure 4.1 The "Register" Design Use Case

Use Case Name:	Change Registration Info	
Actor(s):	User	
Description:	A user can opt to change the personal details given during registration whenever he or she wants to.	
Typical Course of Events:	**Actor Action**	**System Response**
	Step 1: The user clicks the "Change Registration Info" link from any window in the auction site. **Step 4:** The user changes in his/her particulars in the text boxes that are labeled NAME, E-MAIL ADDRESS, PHONE NUMBER, STREET ADDRESS, CITY, STATE and POSTCODE. **Step 5:** After changing the particulars, the user clicks the [Submit] button to submit the form.	**Step 2:** The system checks if the user is logged in. **Step 3:** If the user is logged in, the system retrieves all the relevant personal particulars of the user and displays it in text boxes in a new window. **Step 6:** The system verifies that all the text boxes are filled. **Step 7:** After the verification process, the system updates the record of the object AuctionUser in the database containing all the information submitted by the user. **Step 8:** The system replaces the current window a new window confirming the information update.
Alternate Courses:	**Step 3:** If the user not is logged in, the system invokes the "Login" use case. **Step 7:** If not all the text boxes have been filled, the system displays the user info window again and includes a message notifying the user to re-submit the form. Go to **Step 4**.	

Figure 4.3 The "Change Registration Info" Design Use Case

Use Case Name:	Change Password	
Actor(s):	User, Administrator	
Description:	As a safety feature, users can change his or her password as often as possible through this service provided by the system.	
Typical Course of Events:	**Actor Action**	**System Response**
	Step 1: The user clicks the "Change Password" link from any window in the auction site. **Step 4:** The user keys in the PASSWORD twice. **Step 5:** After keying in, the user clicks the [Submit] button to submit the form.	**Step 2:** The system checks if the user is logged in. **Step 3:** If the user is logged in, the system displays a new window that requires the user to key in his/her new PASSWORD twice in two separate text boxes. **Step 6:** The system verifies that all the text boxes are filled. **Step 7:** After the verification process, the system updates the record of the object AuctionUser in the database containing the new PASSWORD submitted by the user. **Step 8:** The system replaces the current window a new window confirming the PASSWORD update.
Alternate Courses:	**Step 3:** If the user not is logged in, the system invokes the "Login" use case. **Step 7:** If not all the text boxes have been filled, the system displays the user info window again and includes a message notifying the user to re-submit the form. Go to **Step 4**.	

Figure 4.4 The "Change Password" Design Use Case

Use Case Name:	Select Category	
Actor(s):	User	
Description:	A user can browse through the categories available in the auction site to look for items of interest located at a particular category.	
Typical Course of Events:	**Actor Action** The homepage is currently displayed on the screen. **Step 1:** The user clicks on a particular CATEGORY NAME link.	**System Response** **Step 2:** The system identifies that category and displays the items in that particular category in new window. The ITEM NAME, END DATE, END TIME, HITS, NUMBER OF BIDS and CURRENT PRICE is displayed for each item.
Alternate Courses:	**Step 2:** If the database does not have any items in that category, the system displays a new window that contains a message stating "There are no items in this category".	

Figure 4.5 The "Select Category" Design Use Case

Use Case Name:	Search for Item	
Actor(s):	User	
Description:	A user can type in a keyword of the item he or she is interested in and in return the system will display all the relevant items available.	
Typical Course of Events:	**Actor Action** The homepage is currently displayed on the screen. **Step 1:** The user keys in the keywords of the item which he/she wants to search for in the text box labeled "Search". **Step 2:** The user clicks the [Search] button.	**System Response** **Step 3:** The system searches the database for all the matching items entered by the user. **Step 4:** The system displays the items available from the search in a new window. The ITEM NAME, END DATE, END TIME, HITS, NUMBER OF BIDS and CURRENT PRICE is displayed for each item.
Alternate Courses:	**Step 2:** If the cannot find any matches from the search, the system displays a new window that contains a message stating "There are no matches from the keywords".	

Figure 4.6 The "Search for Item" Design Use Case

Use Case Name:	View Item Info	
Actor(s):	User	
Description:	Each item posted in the auction site has some details entered by it seller. A potential buyer can view this information before deciding whether to bid for that item or not.	
Typical Course of Events:	**Actor Action** The window after the initiation of the use cases "Select Category" or "Search for Item" is currently on the screen. **Step 1:** The user clicks on a particular ITEM NAME link.	**System Response** **Step 2:** The system identifies that item and displays the particulars in that particular item in new window. The ITEM NAME, END DATE, END TIME, HITS, NUMBER OF BIDS and CURRENT PRICE is displayed for that item.
Alternate Courses:	None	

Figure 4.7 The "View Item Info" Design Use Case

Use Case Name:	Bid for Item	
Actor(s):	User	
Description:	Once a user makes the decision to bid for an item, the system will provide the necessary interface for the use to make a bid by entering the specific amount of the bid.	
Typical Course of Events:	**Actor Action**	**System Response**
	The window after the initiation of the use case "View Item Info" is currently on the screen.	
	Step 1: The user clicks on a particular "Bid for Item" link.	**Step 2:** The system checks if the user is logged in.
	Step 4: The user enters the bidding amount in the text box that is labeled BID AMOUNT.	**Step 3:** If the user is logged in, the system displays a new window containing the bidding form.
	Step 5: The user clicks the [Bid] button to submit the bidding form.	**Step 6:** The system verifies that the BID AMOUNT is valid.
		Step 7: After the verification process, the system creates a new record of the object AuctionBid in the database containing all the bidding information submitted by the user.
		Step 8: The system replaces the current window a new window confirming the bid.
Alternate Courses:	**Step 3:** If the user not is logged in, the system invokes the "Login" use case. **Step 7:** If the bid amount is invalid, the system displays the bid form again and includes a message notifying the user to re-submit the form. Go to **Step 4**.	

Figure 4.8 The "Bid for Item" Design Use Case

Use Case Name:	Post Item	
Actor(s):	User	
Description:	A user can also post an item for sale by setting a minimum price for that item and determining the increment price for bidding.	
Typical Course of Events:	**Actor Action**	**System Response**
	The window after the initiation of the use case "View Item Info" is currently on the screen.	
	Step 1: The user clicks on a particular "Post Item" link.	**Step 2:** The system checks if the user is logged in.
	Step 4: The user selects an CATEGORY NAME which the AuctionItem belongs to.The user enters the item particulars in the fields that are labeled ITEM NAME, DESCRIPTION, MINIMUM PRICE, INCREMENT, RESERVE PRICE, START DATE, END DATE SHIPPING METHOD, SHIPPING PAID BY AND PAYMENT METHOD.	**Step 3:** If the user is logged in, the system displays a new window containing the post item form. **Step 6:** The system verifies that all the text boxes are filled. **Step 7:** After the verification process, the system creates a new record of the object AuctionItem in the database containing all the item information submitted by the user.
	Step 5: The user clicks the [Post Item] button to submit the post item form.	**Step 8:** The system replaces the current window a new window confirming the item posting.
Alternate Courses:	**Step 3:** If the user not is logged in, the system invokes the "Login" use case. **Step 7:** If not all the text boxes have been filled, the system displays the post item form again and includes a message notifying the user to re-submit the form. Go to **Step 4**.	

Figure 4.9 The "Post Item" Design Use Case

Use Case Name:	Rate a User	
Actor(s):	User	
Description:	After a transaction has been made, a user can rate the seller or buyer according to the user's opinion. This rating method will look into the delivery, payment and accuracy of posting of the seller or buyer.	
Typical Course of Events:	**Actor Action** The window after the initiation of the use case "View Item Info" is currently on the screen. **Step 1:** The user clicks on the "Rate this seller" or "Rate this buyer" link. **Step 4:** The user enters the rating particulars. **Step 5:** The user clicks the [Submit] button to submit the rating form.	**System Response** **Step 2:** The system checks if the user is logged in. **Step 3:** If the user is logged in, the system displays a new window containing the rating form. **Step 6:** The system verifies that all the text boxes are filled. **Step 7:** After the verification process, the system updates the record of the object UserRating in the database. **Step 8:** The system replaces the current window a new window confirming the rating.
Alternate Courses:	**Step 3:** If the user not is logged in, the system invokes the "Login" use case. **Step 7:** If not all the text boxes have been filled, the system displays the post rating form again and includes a message notifying the user to re-submit the form. Go to **Step 4**.	

Figure 4.10 The "Rate a User" Design Use Case

Use Case Name:	View User Rating	
Actor(s):	User	
Description:	Before deciding to sell or bid for an item, users can view the buyer or seller's rating to ensure genuine transactions.	
Typical Course of Events:	**Actor Action** The window after the initiation of the use case "View Item Info" is currently on the screen. **Step 1:** The user clicks on the "View user rating" link.	**System Response** **Step 2:** The system identifies the user and displays the rating info of that particular user in new window.
Alternate Courses:	None	

Figure 4.11 The "View User Rating" Design Use Case

Use Case Name:	Add a Category	
Actor(s):	Administrator	
Description:	The administrator can add a category into the system when there is a need for a new category.	
Typical Course of Events:	**Actor Action** The window after the initiation of the use case "Login" for the administrator is currently on the screen. **Step 1:** The administrator enters the new CATEGORY NAME in a text box. **Step 2:** The administrator clicks on the [Add Category] button.	**System Response** **Step 3:** The system adds a new record of the object AuctionCategory in the database. **Step 4:** The system refreshes the administration menu with the new category displayed.
Alternate Courses:	None	

Figure 4.12 The "Add a Category" Design Use Case

Use Case Name:	Delete a Category	
Actor(s):	Administrator	
Description:	The administrator can delete a category off the system when there is a no need for that category due to poor response or irrelevance.	
Typical Course of Events:	**Actor Action** The window after the initiation of the use case "Login" for the administrator is currently on the screen. **Step 1:** The administrator clicks the check box next to the CATEGORY NAME to be deleted. **Step 2:** The administrator clicks the [Delete Category] button.	**System Response** **Step 3:** The system deletes the particular record of the object AuctionCategory in the database. **Step 4:** The system refreshes the administration menu with the remaining categories displayed.
Alternate Courses:	None	

Figure 4.13 The "Delete a Category" Design Use Case

Use Case Name:	View User Info	
Actor(s):	Administrator	
Description:	The administrator can view all the personal data provided by the user to check ensure only genuine users use the auction site.	
Typical Course of Events:	**Actor Action** The window after the initiation of the use case "Login" for the administrator is currently on the screen. **Step 1:** The administrator clicks on the "View user info" link.	**System Response** **Step 2:** The system identifies the user and displays the user info of that particular user in new window.
Alternate Courses:	None	

Figure 4.14 The "View User Info" Design Use Case

Use Case Name:	Delete User	
Actor(s):	Administrator	
Description:	The administrator can delete a user off the system once that user is found to be non-genuine through poor ratings and complaints made by other users to the administrator.	
Typical Course of Events:	**Actor Action** The window after the initiation of the use case "Login" for the administrator is currently on the screen. **Step 1:** The administrator clicks the check box next to the USER NAME to be deleted. **Step 2:** The administrator clicks the [Delete User] button.	**System Response** **Step 3:** The system deletes the particular record of the object AuctionUser in the database. **Step 4:** The system refreshes the administration menu with the remaining users displayed.
Alternate Courses:	None	

Figure 4.15 The "Delete User" Design Use Case

Use Case Name:	View Report	
Actor(s):	Administrator	
Description:	The administrator can also perform certain queries to the system's database to retrieve information for reporting purposes.	
Typical Course of Events:	**Actor Action** The window after the initiation of the use case "Login" for the administrator is currently on the screen. **Step 1:** The administrator keys in the SQL query to retrieve the information needed. **Step 2:** The administrator clicks the [Get Report] button.	**System Response** **Step 3:** The system processes the query entered by the administrator. **Step 4:** The system displays a new window containing the results of the query processing.
Alternate Courses:	None	

Figure 4.16 The "View Report" Design Use Case

4.2 Object Interactions and Behavior

In the previous part, we refined the use cases to reflect the implementation environment. In this section, we will determine the objects' behaviors and responsibilities by creating detailed models of how the objects will interact with each other to provide the functionality specified in each design use case.

4.2.1 Sequence Diagrams

The UML provides the sequence diagram that graphically depicts interactions between objects. The sequence diagram shows in great detail how the objects interact with each other over time.

The sequence diagram is read from top to bottom, following the logic of the use case. A vertical line symbolizes each object referenced in the use case. A rectangular box on the vertical line represents the behaviors or operations that each object needs to fulfil an obligation. These boxes represent program code. The arrows between the lines represent interactions or messages being sent to a particular object to invoke one of its operations to satisfy a request.

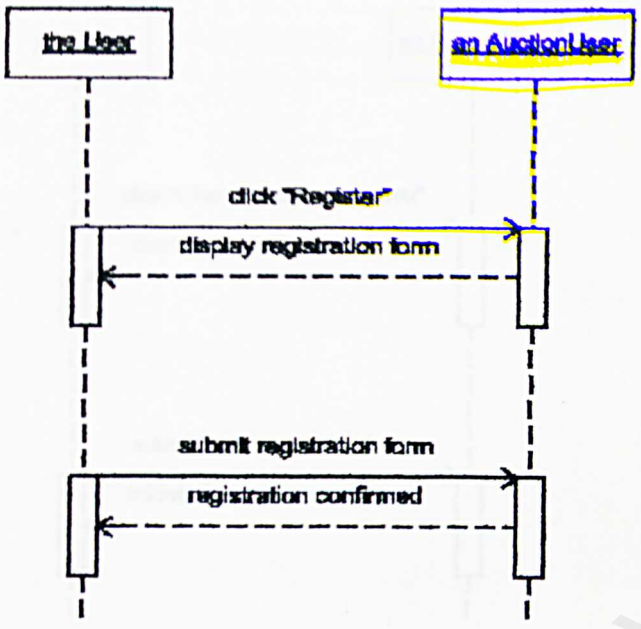


Figure 4.17 Sequence Diagram for "Register" Use Case

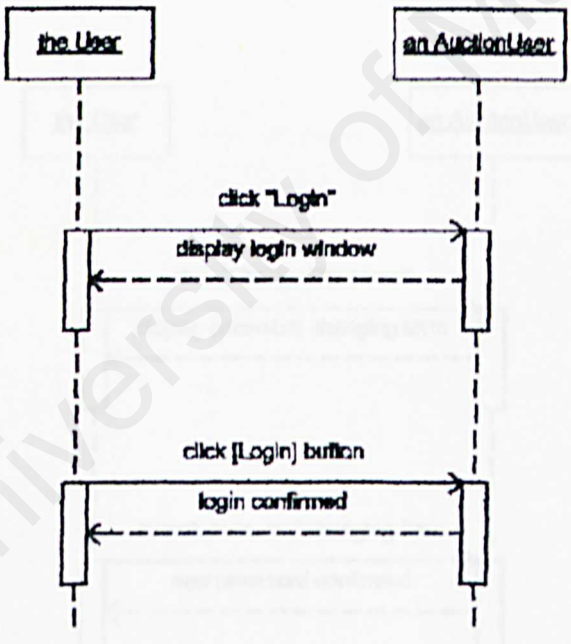


Figure 4.18 Sequence Diagram for "Login" Use Case

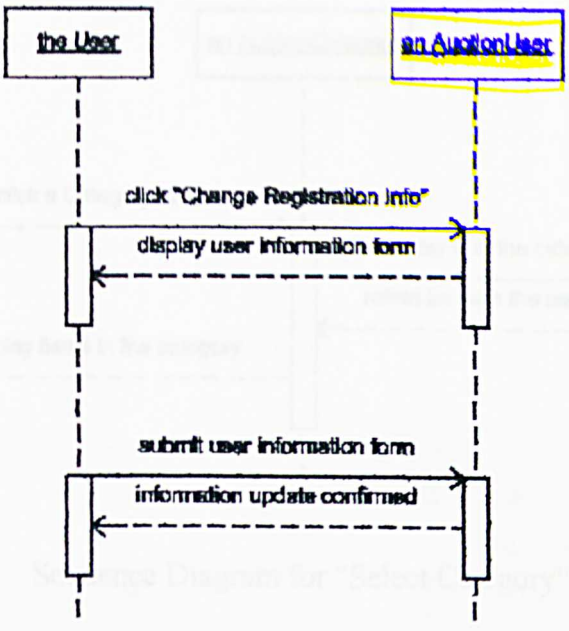


Figure 4.19 Sequence Diagram for "Change Registration Info" Use Case

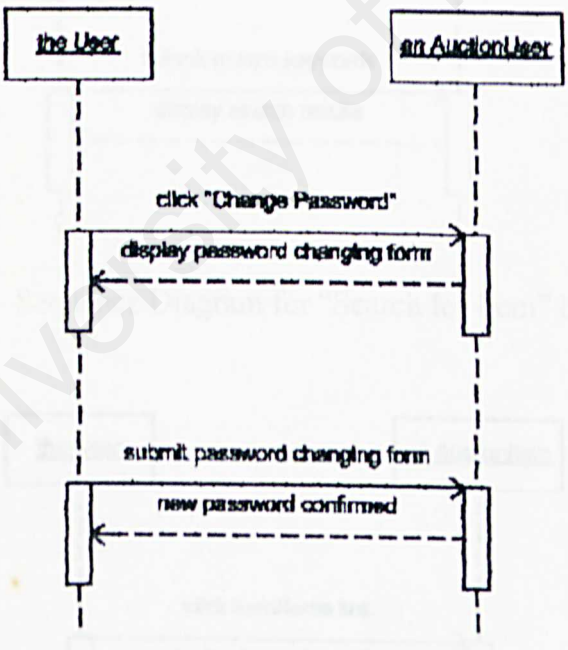


Figure 4.20 Sequence Diagram for "Change Password" Use Case

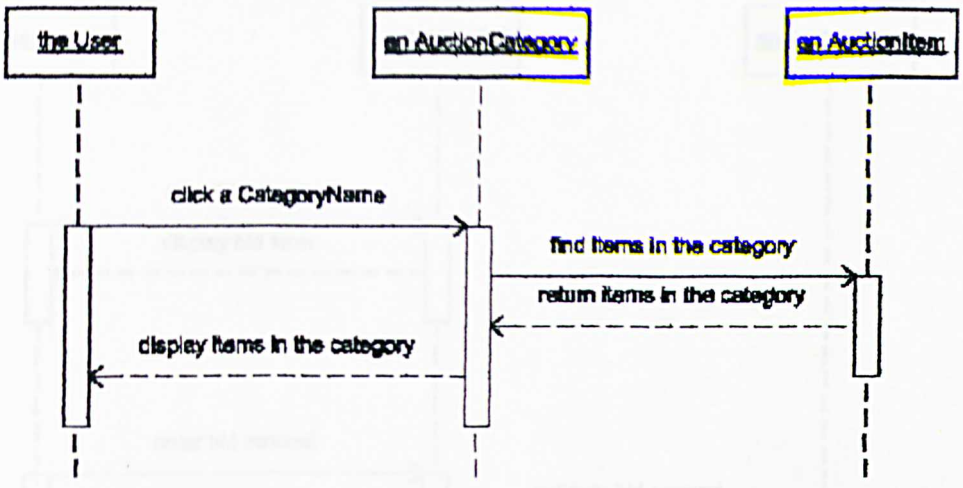


Figure 4.21 Sequence Diagram for "Select Category" Use Case

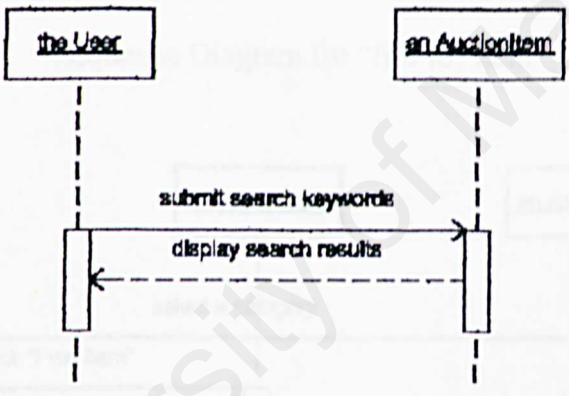


Figure 4.22 Sequence Diagram for "Search for Item" Use Case

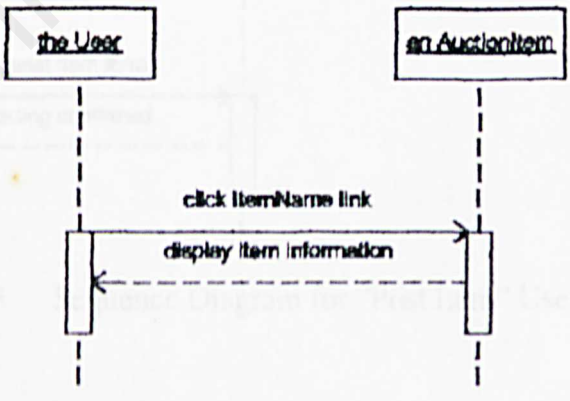


Figure 4.23 Sequence Diagram for "View Item Info" Use Case

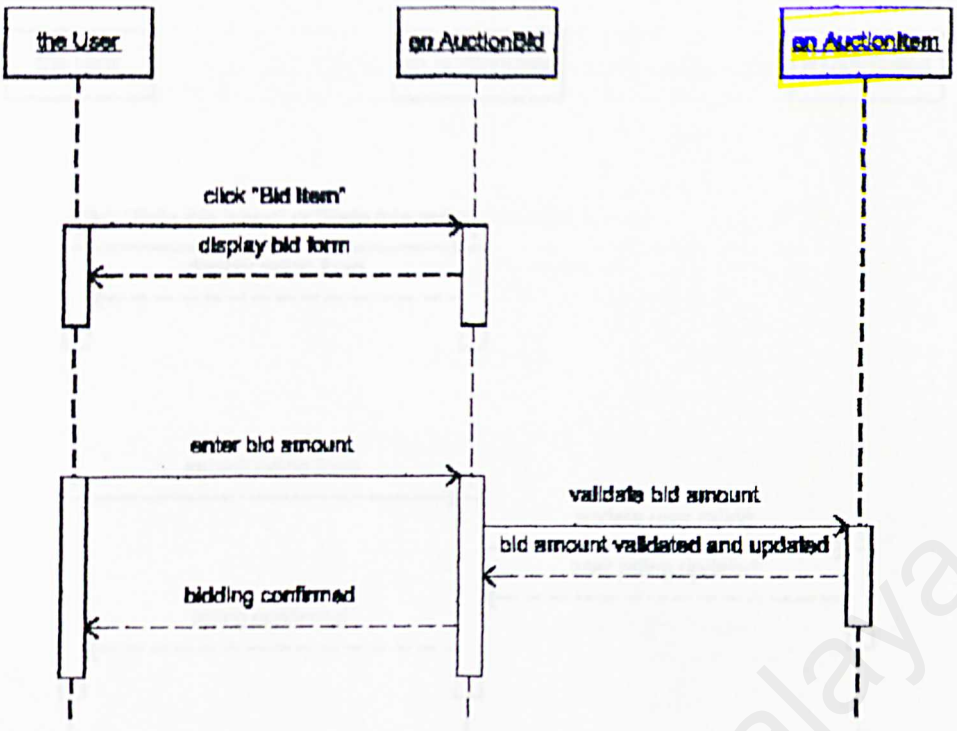


Figure 4.24 Sequence Diagram for "Bid for Item" Use Case

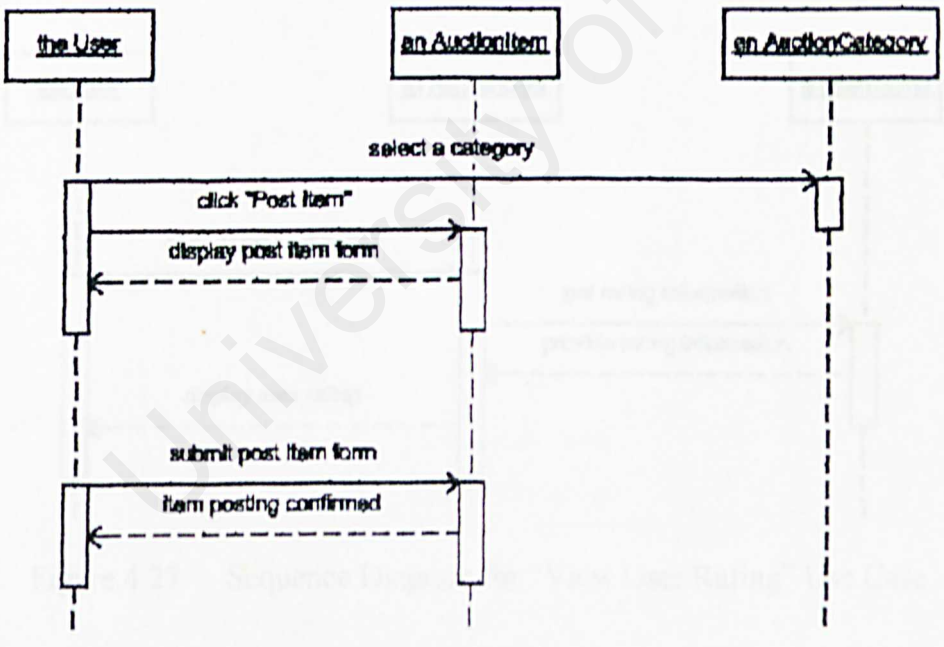


Figure 4.25 Sequence Diagram for "Post Item" Use Case

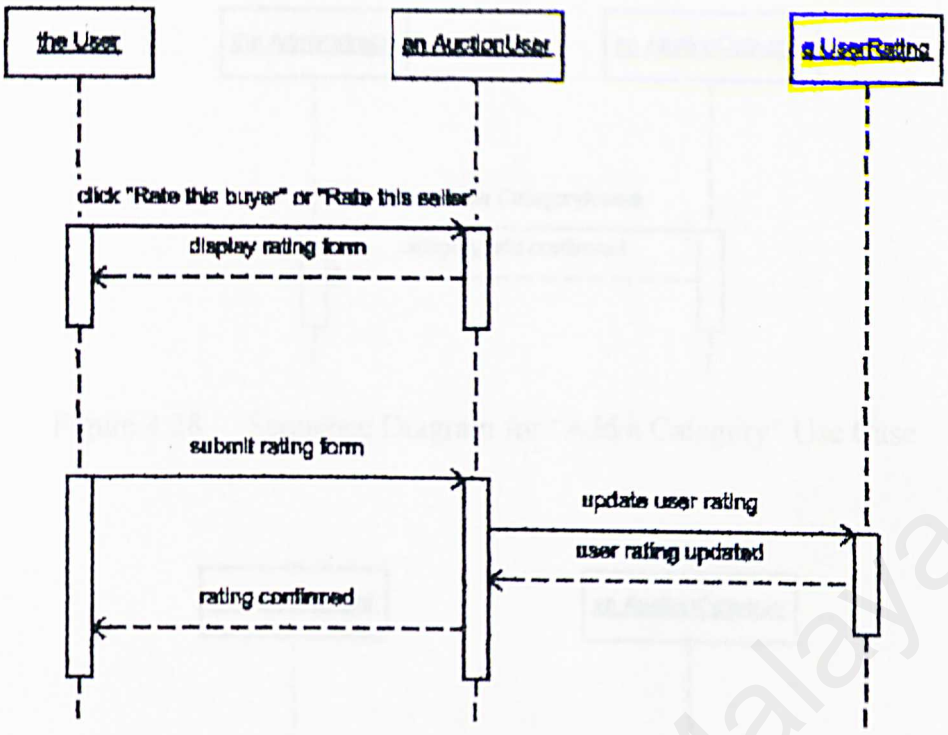


Figure 4.26 Sequence Diagram for "Rate a User" Use Case

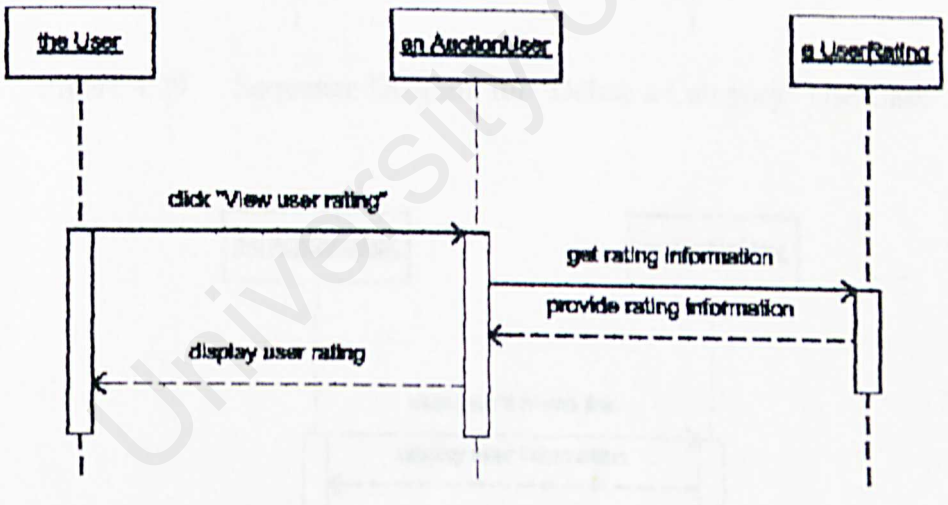


Figure 4.27 Sequence Diagram for "View User Rating" Use Case

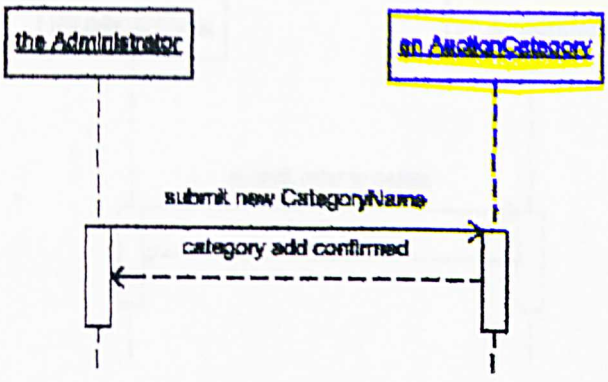


Figure 4.28 Sequence Diagram for "Add a Category" Use Case

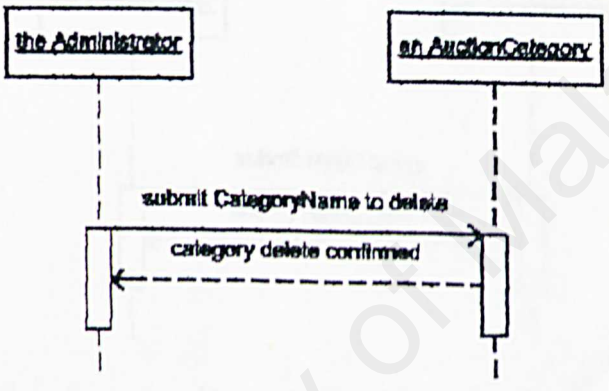


Figure 4.29 Sequence Diagram for "Delete a Category" Use Case

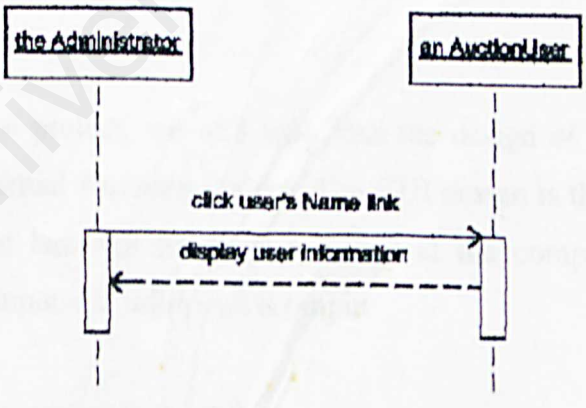


Figure 4.30 Sequence Diagram for "View User Info" Use Case

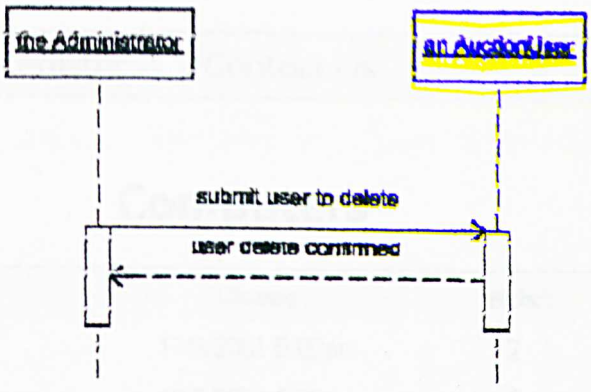


Figure 4.31 Sequence Diagram for "Delete a User" Use Case

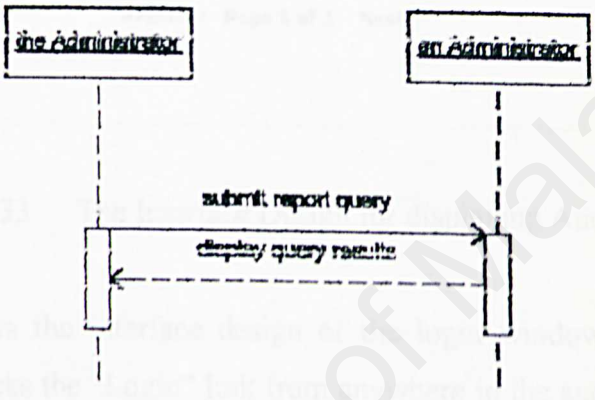


Figure 4.32 Sequence Diagram for "View Report" Use Case

4.3 Interface Design

In this part of the project, we will look into the design of the Graphical User Interface (GUI) of the Virtual Auctions system. The GUI design is the specification of a dialogue or conversation between the system user and the computer. This dialogue generally results in data input and information output.

Figure 4.33 shows the GUI for the page that displays all the AuctionItems in a particular AuctionCategory. This screen will appear after the initiation of the "Select Category" use case.

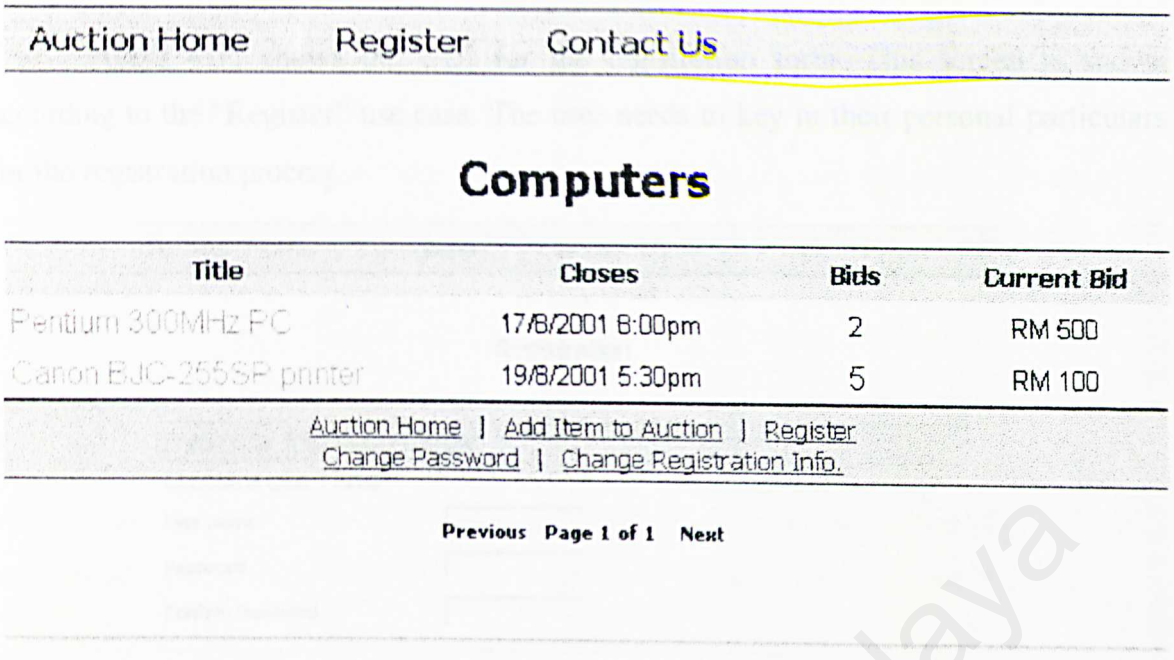


Figure 4.33 The Interface Design for displaying AuctionItems

Figure 4.34 shows the interface design of the login window. This window is shown when the user clicks the "Login" link from anywhere in the auction site. The user is required to enter his or her LOGIN NAME and PASSWORD and click the [Login] button to submit the login information.

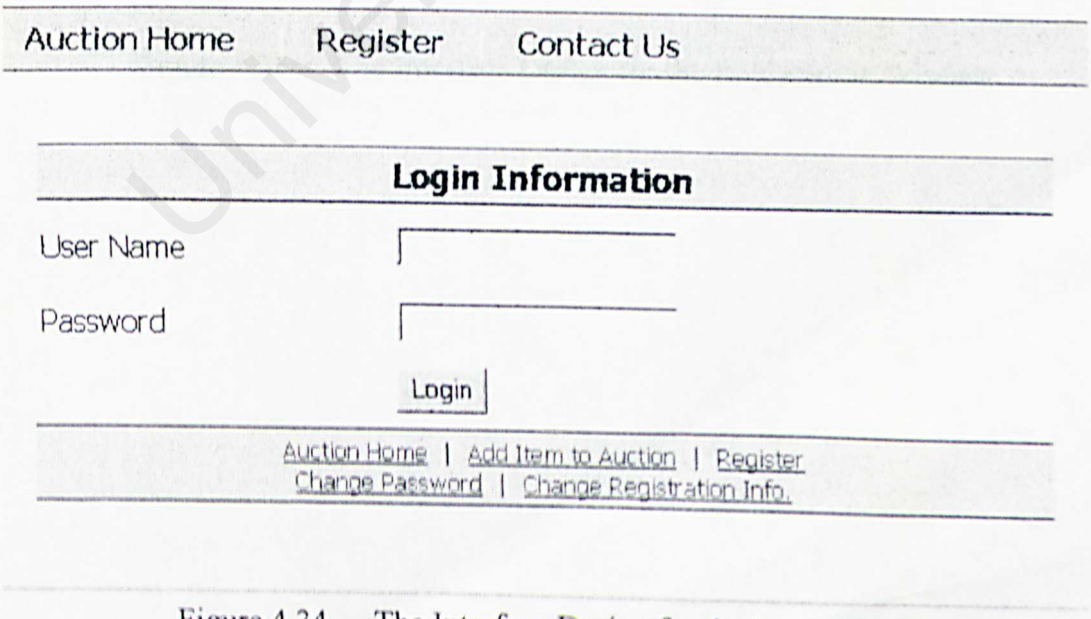


Figure 4.34 The Interface Design for the Login Window

Figure 4.35 shows the GUI for the registration form. This screen is shown according to the "Register" use case. The user needs to key in their personal particulars for the registration process.

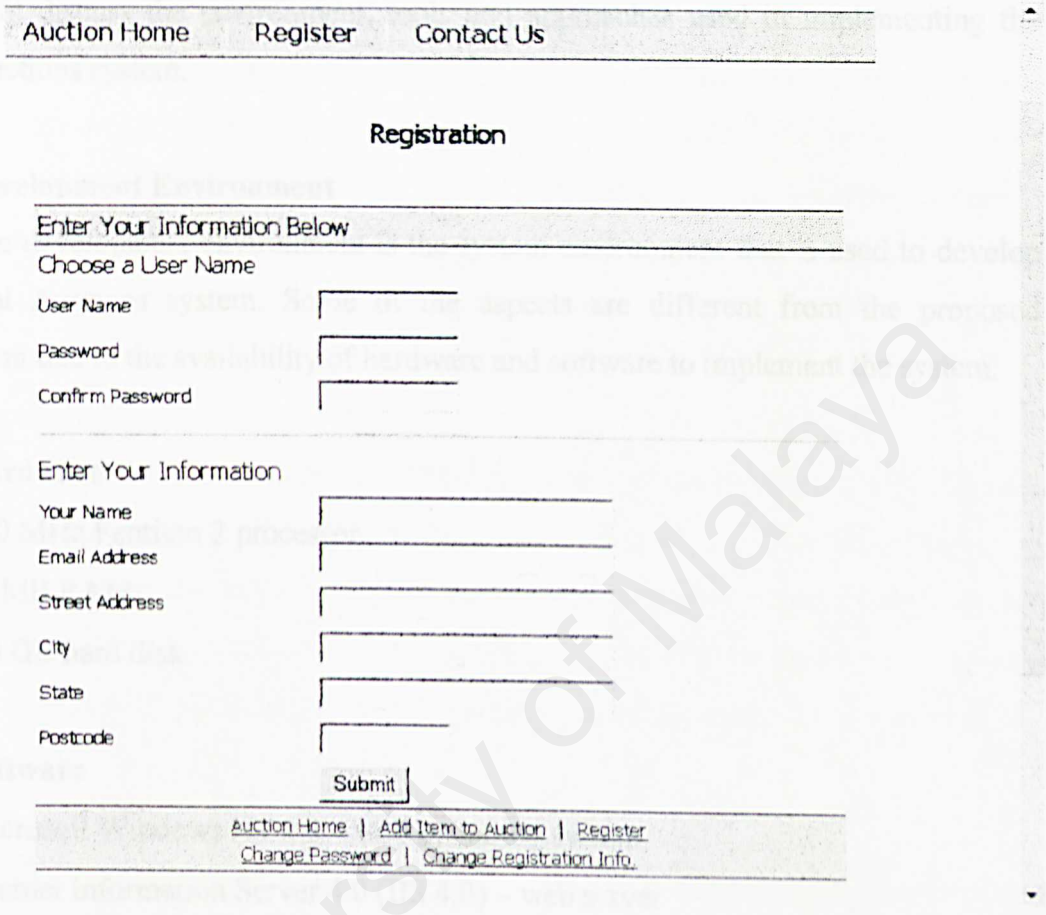


Figure 4.35 The Interface Design for the Registration Window

CHAPTER 5 : SYSTEM IMPLEMENTATION

The Virtual Auctions system is developed using the bottom-up approach. Small modules are developed before being integrated into a complicated functional system. This chapter will discuss the environment, tools and approaches used in implementing the Virtual Auctions system.

5.1 Development Environment

The development environment is the system environment that is used to develop the Virtual Auctions system. Some of the aspects are different from the proposed environment due to the availability of hardware and software to implement the system.

5.1.1 Hardware

- 300 MHz Pentium 2 processor
- 32 MB RAM
- 3.0 GB hard disk

5.1.2 Software

- Microsoft Windows NT 4.0 – web operating system
- Internet Information Server 4.0 (IIS 4.0) – web server
- Microsoft Frontpage – web application development tool
- Microsoft Internet Explorer 5.0 – web browser
- Microsoft SQL Server 7.0 – database server
- Microsoft Word – word processing
- Microsoft Visio Professional – object oriented modelling

5.2 Coding

5.2.1 Programming Language

The Virual Auctions system is developed using ASP 3.0 web programming technology. This programming technology has many similarities with the PHP and Cold Fusion where the script is embedded in the HTML document. The ASP document has a ".asp" extension and the scripts can be inserted anywhere in the document between the "<%...%>" tags.

There are two types of scripts used in the Virtual Auctions system; the client-side scripts and server-side scripts. Client-side scripts are executed by the desktop broeser whereas server-side scripts are executed by the web server before the page content is passed on to the calling browser. In this system, Javascript was used for client-side scripting and Vbscript was used or server-side scripting.

5.2.2 Database

The database server used in the Virtual Auctions system is Microsoft SQL Server 7.0. The approach used to connect to the database is through the Connection object of ASP. After a connection has been made using the Connection object, ASP uses the Recordset object to access the data in the database. Through these two built-in objects made available in ASP, the database server is easily accessible using SQL statements.

5.2.3 Approach

The coding approach in developing the Virtual Auctions system is using iterative and incremental development. The coding starts with the development of reusable fundamental methods and classes to facilitate the follow up development. These scripts are designed specifically for the Virtual Auctions system to run smoothly regardless of the implementation environment.

5.2.4 Documentation

Documentation plays a very important role in all software developments including web-based applications such as the Virtual Auctions system. In this project, the documentation developed is mainly grouped into:

- User "Frequently Asked Questions (FAQ)" help files – embedded into the Virtual Auctions system where it can accessed from the "FAQ" hyperlink made available in every page.
- Admistrator manual – a hard copy document for the reference of the site administrator.
- Comments in coding – descriptions inserted between HTML coding and ASP scripts where applicable to further describe the statements, which are more ambiguous.

CHAPTER 6 : SYSTEM TESTING

In order to verify that the Virtual Auctions system has satisfied the proposed requirements and to identify the differences between the expected and actual results, a series of tests are carried out. The testing stage aims to uncover bugs which can be any unexpected, questionable, or undesired aspects or behaviour displayed or caused by the system. Testing can uncover the different classes of errors in a minimum amount of time and effort if it is done in a proper way. The testing strategies used in the development of Virual Auctions include unit testing, integration testing and system testing.

6.1 Testing Techniques

There are two techniques used in testing which in either one has their own advantages and disadvantages. The White Box testing is used in the unit testing and Black Box testing is more suitable for system testing. The choice of testing techniques depends on many factors, including:

- The number of possible logical paths
- The nature of the input data
- The amount of compilation involved
- The complexity of algorithms

6.1.1 White Box Testing

Sometimes named Open Box Testing or Clear Box Testing, White Box Testing allows us to use the structure of the test module to test in different ways. For example, we can derive test cases that execute all statements or all the control paths within the components to be sure that the object is working properly. White Box testing has the ability to test:

- All logical decisions
- Loop boundaries
- Data structures to assure internal validity

- Guarantee all independent paths within module have been exercised at least once

6.1.2 Black Box Testing

Unlike White Box testing, Black Box testing is more favoured to ensure that every kind of input is submitted, and the output observed matches the output expected. It is also called Closed Box testing, where the testing is performed in the situation that the contents are unknown. Testing begins with the feeding input to the tested system and notes what output is produced. Black Box testing is able to uncover various types of errors include:

- Interface error
- Incorrect or missing functions
- Performance errors
- Initialisation and termination error

6.2 Testing Strategies

6.2.1 Unit Testing

Unit testing is performed concurrently with the coding phase. It is a White Box testing method where all codes are reviewed to ensure its correctness and validity. The major Elements in unit testing includes:

- Examine the program code by reading through it, trying to spot algorithm, data and syntax faults.
- Comparing the code with specificationsand with the design to make sure that all relevant cases have been considered.
- The result web page is generated and remaining syntax faults are uncovered and eliminated if necessary.
- Test cases are developed to show that the input is properly converted to the desired outpuut.

6.2.2 Integration Testing

Integration testing ensure that all units developed and tested can work together. In this stage, system is viewed as a hierarchy of components and the components are tested before combined for further testing. There are two key approaches in integration, which are bottom-up integration and top-down integration. Bottom-up integration tests individually all the components at the lowest level of the system hierarchy follwed by the components that call the previously tested components.

This process is performed repeatedly until all the are included in the testing. Top-down integration testing starts with the component at the top-most level. Then all components called by the tested components are combined and tested as a larger unit. This approach is reapplied until all components are incorporated. In the case of the Virtual Auctions system, the bottom-up integration approach is used.

6.2.3 System Testing

System testing ensures the system does what the user desires it to do. Unlike unit and integration testing, which are intended to fulfil the design specifications, system testing focuses on the fulfilment of the functional requirements, non-funtional requirements as well as the user requirements and environments. System testing involves users of various groups to evaluate the system in order to ensure all parts of the system are tested.

6.3 Analysis of Test Results

6.3.1 Main Objectives of Project Achieved

According to response and results of testing, the system goes in line with the project's objectives. The system has been loaded with all the functions proposed and generally is user friendly.

6.3.2 Graphics Upload Not Supported

Users of the system especially sellers are not given the option to upload the pictures of the item to be auctioned. Graphics upload is not only found to be important to improve the possibility of an item being sold but also to improve the attractiveness of the auction site.

6.3.3 Enhancement of Report Generation

The report generation part of the system is not dynamic enough for the administrator to retrieve the exact information needed from the database. Currently the report generation page only sums up the total of categories, items, users and bids currently on the system.

CHAPTER 7 : CONCLUSION

This chapter generally focuses on the final outcome of the Virtual Auctions system and the conclusion on test results.

7.1 Goals Achieved

7.1.1 Objectives of Virtual Auctions Achieved

In general, all the objectives of the Virtual Auctions system are achieved. However certain use cases discovered in the analysis part of the project was not developed due to time constraints. An overall view of the system will reveal that the system is indeed user friendly and will be able to draw all levels of users in the Internet. Besides that, the system is found to be very reliable and the response time is fast.

7.1.2 Foundation for Future Growth and Enhancement

The implementation of the Virtual Auctions system has proven to be very capable in terms of scalability and flexibility for future upgrading and enhancements. New functional systems can be added into the Virtual Auctions system as well as enhancementscan be made to the current system without disrupting othe components.

7.2 Goals Not Achieved

7.2.1 Database Problems

The functions of the SQL Server are not fully utilised as no relationship mappings are made between the related tables in the database. This is because the tables are only related through the utilisation of triggers set in each table to facilitate cascade deletion and to maintain referential integrity.

7.2.2 Use Cases Left Out

Certain use cases discovered in the analysis part of the project was not developed due to time constraints. For example the "Search for Item", "Rate a User" and "View User Rating" use cases was left out the development process and needs future upgrading to be fulfilled.

7.2.3 Report Generation Weakness

The report generation part of the system is not dynamic enough for the administrator to retrieve the exact information needed from the database. Currently the report generation page only sums up the total of categories, items, users and bids currently on the system. Therefore the objective of creating a paperless admionistration system with easy access to the system's database is not fully achieved.

7.3 Application Strength

7.3.1 User Friendly

The system is generally user friendly. The users are able to use the auction system without proper guidance using manuals, courses, etc. The Frequently Asked Questions (FAQ) page integrated into the system fulfils all user questions about the system from Registration to the Bidding process.

7.3.2 Easy Accessibility

The system is a web-based application and can be accessed easily using a standard web browser from any location of the world.

7.3.3 Easy Administration

The adminisration features found in the Virtual Auctions system is indeed easy to use. The User Management feature allows the administrator to view all the user's information and delete user found to be not genuine. The same goes for for Category management and Item Management.

7.4 Application Limitation

7.4.1 Lack of Graphic Support

Users of the system especially sellers are not given the option to upload the pictures of the item to be auctioned. Graphics upload is not only found to be important to improve the possibility of an item being sold but also to improve the attractiveness of the auction site.

7.4.2 Report Generation Facility

The administrator is not given the freedom to extract information that he/she wants to from the database. The report generation facility is not dynamic enough to let the administrator enter his/her own query. Therefore there is a limitation in the report generation facility.

7.5 Difficulties Encountered and Solutions

7.5.1 Referential Integrity Problem

The referential integrity problem is a condition in the database where interrelational constraints must be satisfied. This means that if a record is deleted from a table, other records in other tables related to that particular table must be deleted also. Therefore this problem is solved using cascade delete triggers set in the SQL Server. A confirmation window pops up each time an administrator wants to delete a record to remind him/her that the deletion process may trigger cascade deletion on other tables.

7.5.2 Item Current Price Inconsistency

The current price of an item remained the same even if the latest bid was deleted due to user deletion by the administrator. Therefore, the problem was found to be caused by redundant data in the tblAuctionItem table. The current price of an item actually does not need to be stored as an item's attribute. It can be generated using an SQL query to find the highest bid amount of the item.

7.6 Knowledge, Skill and Experience

The Virtual Auctions project has introduced me to the world of web programming. I had the opportunity to experience web server and database administration and configuration. Furthermore, by doing the literature survey, I gained much knowledge on the various considerations of wep application development from various aspects such as platforms, programming choices and web architectures.

Besides that, I had the opportunity to learn new scripting languages such as Vbscript and Javascript. Lastly, working on the SQL Server as the database provider introduced me to the many features of a full-scale database server.

BIBLIOGRAPHY

[DEI99] Deitel, P., Deitel, H., Nieto, T. "e-Business & e-Commerce How To Program": Prentice Hall, 1999.

[LEL01] "www.lelong.com.my", 2001

[NOR99] Norton, P. "Peter Norton's Introduction To Computers": Glencoe/McGraw – Hill, 1999.

[DIX96] Dixon, R. "Client/Server And Open Systems": John Wiley & Sons, Inc., 1996.

[RES00] Reselman, B. "Active Server Pages 3.0 By Example": QUE, 2000.

[MIC01] "www.microsoft.com", 2001

[SOB95] Sobell, M. "A Practical Guide To The UNIX System": Addison-Wesley, 1995.

Whitten, J., Bentley, L., Dittman, K. "System Analysis And Design Methods": McGraw-Hill, 2000.

Kroenke, D. "Database Processing – Fundamentals, Design and Implementation": Prentice Hall, 2000.

Appendix A FREQUENTLY ASKED QUESTIONS (FAQ)

- **How do I Register?**

This auction services is provided to members for FREE but we do require that all sellers and buyers to be registered.

This enables the system to keep track of items for sale and who bids on what. It protects both us as services providers and you as the buyer and/or seller. Registration is simple and painless. Merely click on the Register on the navigation menu to begin the registration process.

- **Types of Auctions**

A. Standard Auction (For Sale)

Standard auction is the most common type of auction where seller posts an item for a limited period of time. Bidders will place their bid on the item. When the period of auction ends, the highest bidder wins the auction.

B. Reverse Auction (Want to buy)

Buyers can post items they want to buy with the price they are willing to pay. Seller who has the item will then make a bid to the interested buyer by decreasing the current price.

- **How do I place a Bid?**

After browsing through the auction site, if you find an item you would like to have, then place a bid (after you register, if you haven't already done so) for the item. This is done by clicking on the "Bid Now" button located beneath the item description. Enter the offer price that you are willing to pay.

- **What are bid Increments?**

When you place a bid, there is a fixed amount by which you have to exceed the last bid that was placed. This is known as a bid increment. Example: Mr. X has bid RM50 for a wallet, and the bid increment set by the seller is RM5. The lowest you can bid, due to the increment of RM5, is RM55 or above.

- **What is a Reserve Price?**

The reserve price is the lowest price a seller is willing to sell an item for. This amount is generally higher than the minimum bid. In order to win the auction, a bidder must meet or exceed the reserve price set by the seller and have the highest bid. If no bidders meet the reserve price, neither the seller nor the highest bidders are under any obligation to complete the transaction.

- **How do I collect my Items and make Payment?**

After the auction is over, arrangements need to be made between the two parties (seller and buyer) for collection and payment. It is the duty of the two, to get in touch with each other. Depending on shipping and payment terms that were stipulated the transaction should proceed as it was stated.

When the auction ends, the email address of the buyer and seller will be available on that particular item description for 2 days. Please initiate the communication without further delay. E-mail the seller to start with, then if necessary, request for phone contact details.

- **How do I pay the Seller?**

Depending on the terms stated on the item description page, or any form that you and the seller have subsequently agreed upon. Please make sure you know the seller well before you make any payment.

- **The Seller refuses to sell the Item to me. What now?**

If the seller refuses to sell the item to you, despite you being the winner of the auction by legitimate means, then let us know. We will pursue the issue and possibly suspend or blacklist the member from using our services, to avoid the incident from recurring.

- **How do I Sell/Buy?**

With an on-line auction site, items are posted for sale and to be looked for on the site, and other members who are looking for something to buy or sell will browse and place

bids for the items that you post. You have to register in order to post items. Login to your account and from there, you can post your items.

- **What can I Sell?**

You are allowed to sell virtually anything. The only conditions that are present are that:

1. Copyrights, patents, trademarks and proprietary rights are not infringed upon.
2. Items are not illegal, or violate and applicable law, stature, ordinance or regulation.
3. Live animals, plants and food items or alcoholic beverage are not posted.

- **How do I change my Password?**

You can change your login password by the following steps:

1. Merely click on the "Change Password" link on the navigation menu to begin the registration process.
2. Enter your Login name, old Password and the new Password you wish to change.
3. If you have forgotten your password, please click on the "Contact Us" link and notify the Administrator.

- **How do I update my Address, Phone Number, etc.?**

Update your personal information by visiting 'Change Registration Info' page by clicking on the "Change User Info" link on the navigation menu.

Appendix B ADMINISTRATOR MANUAL

To access the maintenance page of the Virtual Auctions system, the administrator needs to log in with the adminstrator login name and password:

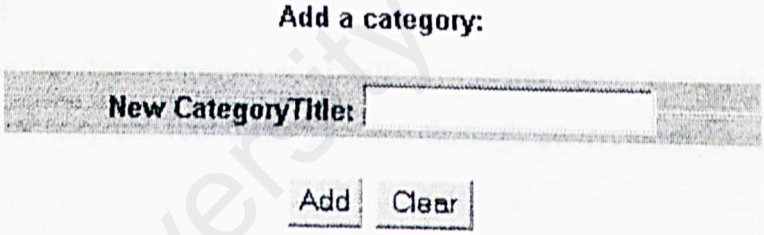
- Login Name : **administrator**
- Password : **admin**

If you want to enter the maintenance page, just use the login name and password provided above. Please take note that this login name and password is hardcoded into the codes. Therefore, it cannot be changed unless you open up the "checklogin.asp" file and change the login information.

Category Management

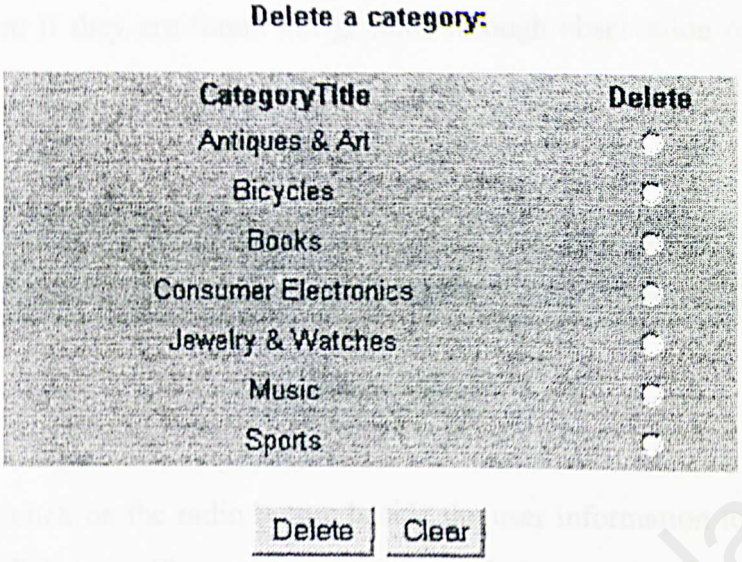
You can use the category management facillity to add and delete categories currently available in the auction system.

- Adding an AuctionCategory:



To add a category, simply enter the new category title into the field provided and click the "Add" button. The new category will automatically be inserted into the system and can be used immediately.

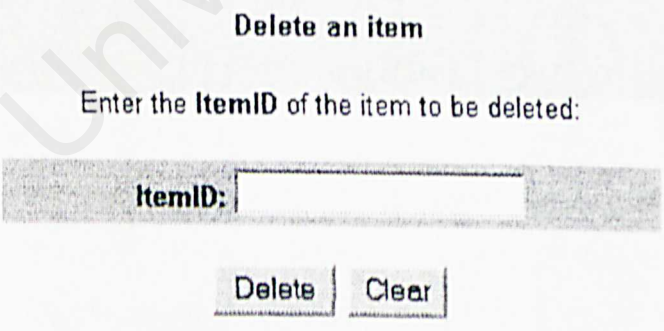
- **Deleting an AuctionCategory**



To delete an auction category, click on the radio button beside the category title to be deleted and click the "Delete" button. Please take note that deleting an AuctionCategory may cause cascade deletion on AuctionItems and AuctionBids.

Item Management

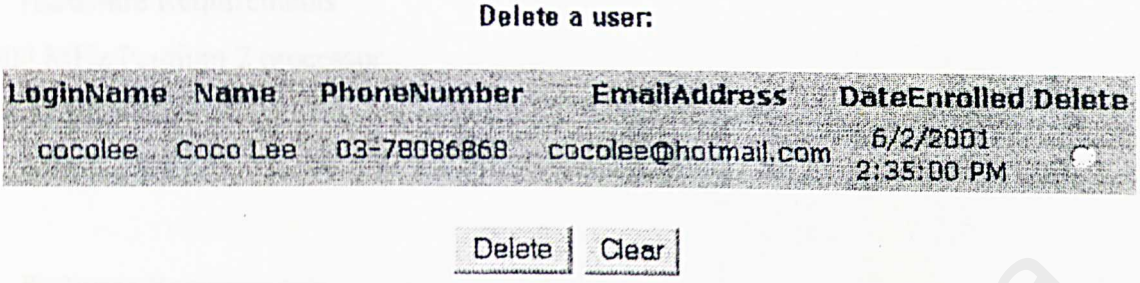
You can use the item management facillity to delete items currently available in the auction system if they are found not genuine through observation or complaints by users.



To delete an item, simply enter the item ID into the field provided and click the "Delete" button. The item will automatically be deleted from the system. Please take note that deleting an AuctionItem may cause cascade deletion on AuctionBids.

User Management

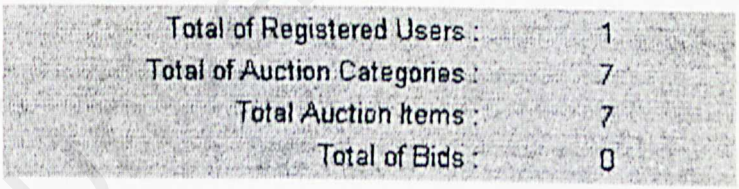
The item management facillity can be used to delete users currently registered in the auction system if they are found not genuine through observation or complaints by users.



To delete a user, click on the radio button beside the user information to be deleted and click the "Delete" button. Please take note that deleting an AuctionUser may cause cascade deletion on AuctionItems and AuctionBids.

Report Generation

The report generation facility in the Virtual Auctions system can generate 4 outputs, which gives the total of categories, items, users and bids currently available in the system. It then displays this information to the administrators.



Appendix C SYSTEM INSTALLATION PROCEDURES

System Environment Requirements

- Hardware Requirements

· 300 MHz Pentium 2 processor

· 32 MB RAM

· 3.0 GB hard disk

- Software Requirements

· Web server : Windows NT Server (IIS 4) or Windows 2000 Server (IIS 5)

· Database Server : MS SQL Server 7 or MS SQL Server 2000

System Installation

Copy the contents of the Virtual Auctions folder into your web server's root directory. Your root directory should then contain 35 files and 1 folder used by the Virtual Auctions system to run properly.

Database Setup

- Connection String

In order to access the information in the database, you need to set a few parameters in the "connstr.inc" file. Open the "connstr.inc" file with a text editor. Enter the database server name, database name, user ID and password in the provided spaces. By default, the connection string will look like this:

```
provider=SQLOLEDB;" & "data source=YourDatabaseServerName;" & "Initial
Catalog=YourDatabaseServerName;" & "user ID=YourDatabaseUserID;" &
"password=YourDatabasePassword;
```

After you have set the required parameters, save the file and close it. It is advisable not to open the other files to avoid messing up with the source codes.

- Data Import

The database tables in provided are in .mdb (MS Access) form. You will need to import the tables from VirtualAuctions.mdb into your database. Open the Enterprise Manager of your SQL Server and use the 'Import Data' facility to import the tables. Your database should then have 4 tables (tblAuctionCategory, tblAuctionItem, tblAuctionUser and tblAuctionBid) to run the system properly.

- Triggers Setup

To maintain referential integrity in the database, you will need to set 3 triggers that are cascade delete triggers. The 3 tables that need the triggers are tblAuctionCategory, tblAuctionItem and tblAuctionUser. Just copy the cascade delete coding from the "triggers.txt" file into the 'Managing Triggers' facility provided by the SQL Server for each of those tables.

Testing

Now that the web server and database has been set, you can test the Virtual Auctions system by using a web browser to connect to the web server's root directory.